10
BLAST

10.1 Introduction

BLAST (Basic Local Alignment Search Tool) is a widely used method for assessing which nucleic acid or protein sequences in a large database have significant similarity to a given query sequence. Many of the results derived in previous chapters, particularly those relating to the maximum of several random variables, the geometric-like distribution, P-values, the renewal theorem, random walks, and sequential analysis, were presented because they are needed in the statistical theory associated with the BLAST procedure described in this chapter. For concreteness the discussion is in terms of protein (amino acid) sequences; the analysis for nucleic acid sequences is similar to that for protein sequences.

Currently there are two implementations of BLAST, one by NCBI (the US National Center for Biotechnology Information) and the other at Washington University. For most of this chapter we consider a simple early version of BLAST, leading to a readily understood statistical analysis. We first describe Washington University's version 1.4, which was used to generate the examples of Section 10.5, and then describe various generalizations leading to the current implementations.

10.2 The Comparison of Two Aligned Sequences

10.2.1 Introduction

We start by considering as given an ungapped global alignment of two protein sequences, both of length N, as shown, for example, in (7.2). This is done mainly as a preliminary step to the generalizations in the following sections. In particular, the generalization to finding the best among all local alignments of two sequences will be considered in Section 10.3. The further generalization to database searches will be considered in Section 10.4.

The null hypothesis to be tested is that for each aligned pair of amino acids, the two amino acids were generated by independent mechanisms, so that if amino acid j occurs at any given position in the first sequence with probability p_j and amino acid k occurs at any given position in the second sequence with probability p'_k, the null hypothesis probability that they occur together in a given aligned pair is

$$\text{null hypothesis probability of the pair } (j,k) = p_j p'_k. \tag{10.1}$$

The theory of Chapter 9 shows that classical statistical testing theory requires the specification of an alternative hypothesis. For the moment we simply write

$$\text{alternative hypothesis probability of the ordered pair } (j,k) = q(j,k), \tag{10.2}$$

without any particular specification of the form of the function $q(j,k)$. The choice of the form of this function is discussed at length in Section 10.2.4.

10.2.2 The BLAST Random Walk

In this section and the following sections we give the basic idea behind the statistical aspects of BLAST, considering first the case described above, of two aligned sequences, both of length N.

We number the positions in the alignment from left to right as positions $1, 2, \ldots, N$. A score $S(j,k)$ is allocated to each position where the aligned amino acid pair (j,k) is observed. The choice of the scores $S(j,k), j,k = 1, 2, \ldots 20$ is discussed in Section 10.2.4: For the moment we note only that there is a close connection between the choice of the choice of the scores $S(j,k)$ and the choice of the alternative hypothesis probabilities $q(j,k)$ given in (10.2).

The matrix $S = \{S(j,k)\}$ is the substitution matrix of the process: aspects of these matrices were discussed in Section 6.5. It is required in the theory, and is assumed throughout, that at least one element in the substitution matrix be positive and, for reasons discussed below, that the null hypothesis mean score $\sum_{j,k} p_j p'_k S(j,k)$ be negative. In order to apply the theory of Chapter 7 we also assume throughout that the greatest common divisor of the scores is 1.

An accumulated score at position i is calculated as the sum of the scores for the various amino acid comparisons at positions $1, 2, \ldots, i$. As i increases, this accumulated score undergoes a random walk, as described for example in (7.3) and Figure 7.2 for the protein sequence comparison given in (7.2). When the null hypothesis is true, the walk has negative drift and will go through a succession of increasingly negative ladder points, as defined in Section 7.1. Because the substitution matrix will usually include elements, or scores, whose values are -2 or less, the accumulated score at any ladder point will not necessarily be one less than the accumulated score at the preceding ladder point. This implies that, in random walk terms, boundary overshoot can occur. An analysis of this overshoot is needed for BLAST calculations, as outlined briefly below in Section 10.2.3.

Let $Y_1, Y_2, \ldots$ be the respective maximum heights of the excursions of this walk relative to the height of any ladder point after leaving this ladder point and before arriving at the next, or relative to the height of the last ladder point and arriving at the end of the sequence. We define $Y_{\max}$ as the maximum of these maxima: $Y_{\max}$ is in effect the test statistic used in BLAST, so it is necessary to find its null hypothesis distribution.

The various random variables $Y_1, Y_2, \ldots$ are independent, and ignoring end effects for now, can be taken as being identically distributed. The asymptotic probability distribution of any Y_i was shown in Chapter 7 to be the geometric-like distribution (7.63). The values of C and λ in this distribution depend on the substitution matrix used and the amino acid frequencies $\{p_j\}$ and $\{p'_k\}$. The probability distribution of $Y_{\max}$ then follows from the theory of Section 2.11, which, apart from C and λ, depends on the mean number of ladder points in the walk. (In the notation of Section 2.11, this is the value of n.) In the following section we discuss the computation of the central parameters C and θ^* as well as the mean number of ladder points, drawing on the random walk theory developed in Chapter 7.

The above procedure shows why it is necessary that the mean score $\sum_{j,k} p_j p'_k S(j,k)$ be negative. If this were not so the BLAST random walk would contain arbitrarily long upward excursions from ladder points and the entire testing procedure would break down.

10.2.3 Parameter Calculations

The expression for C is given in equation (7.61), and requires only notational amendments for application to BLAST. The step size is identified with a score $S(j,k)$, and the null hypothesis probability of taking a step of any size is found from the two sets of frequencies $\{p_j\}$ and $\{p'_j\}$.

The computation of λ also follows the random walk principles laid down in Chapter 7. As noted below equation (7.42), λ $(= \theta^*)$ is found from an equation involving the mgf of the step size in this random walk. When the null hypothesis is true, this equation is

$$\sum_{j,k} p_j p'_k e^{\lambda S(j,k)} = 1. \tag{10.3}$$

The calculation of λ from this equation will usually require numerical methods: See Appendix B.15.

The calculation of the null hypothesis probability distribution of $Y_{\max}$ depends not only on C and λ but also on the mean number of ladder points in the BLAST walk. This mean number depends in turn on the mean distance A between ladder points. A general formula for A is given in equation (7.41) and is readily converted to the situation discussed here. However, the arguments leading to this formula do not necessarily provide an efficient general formula for finding the constants R_{-j} in equation (7.41), and we now describe two alternative approaches.

The first alternative approach uses a decomposition of paths. Consider as a simple example a walk in which the possible steps are $+1$ and -2, with respective probabilities p and $q = 1 - p$. Any ladder point reached in the walk is at a distance 1 or 2 below the previous one. The respective probabilities of these two cases are denoted by R_{-1} and $R_{-2} = 1 - R_{-1}$, as in Chapter 7.

The probability that -2 is a ladder point is the probability that the walk goes immediately to -2, together with the probability of the event that the walk first goes to $+1$, and then starting from $+1$, reaches 0 as the first point reached below $+1$ and then -2 as the first ladder point below 0. This implies that

$$R_{-2} = q + p(1 - R_{-2})R_{-2}. \tag{10.4}$$

The positive solution of this equation is

$$R_{-2} = \frac{-q + \sqrt{4pq + q^2}}{2p}. \tag{10.5}$$

From this the value of R_{-1} follows as $1 - R_{-2}$, and then the value of A follows from equation (7.41).

For general substitution matrices this method might not be effective. In such a case, Karlin and Altschul (1990) provide rapidly converging series expansions that give accurate values of A using only a few terms in the series. We assume from now on that a value of A, arrived at by one method or another, is in hand.

Since the two sequences compared are each of length N, and the mean distance between ladder points is A, the mean number of ladder points is equal for all practical purposes to N/A. While various approximations are involved with this calculation, the intuitive interpretation is clear: If, for example the length N is 1000 and the mean distance A between successive ladder points is 50, one expects about 20 ladder points in the walk involved with the comparison of the two sequences, and this is the value given by the expression N/A.

10.2.4 The Choice of a Score

So far, we have taken the score $S(j,k)$ as given, and have not discussed what might be a reasonable choice, on statistical and genetical grounds, for this score. In applications of BLAST this score, whether found by a BLOSUM or a PAM matrix, is a log likelihood ratio (as discussed briefly in Section 6.5), and we now indicate why this is appropriate.

The random walk described in Section 10.2.2 is determined by the sum of the scores $S(j,k)$ at each position during the walk. In sequential analysis one also considers the sum of scores. In sequential analysis the score used is a log likelihood ratio, arrived at through statistical optimality methods. Specifically, if the random variable Y whose probability distribution is being assessed is discrete, this is the "score" statistic $S_{1,0}(y)$, defined in equation (9.56) as the log likelihood ratio

$$S_{1,0}(y) = \log \frac{P(y;\xi_1)}{P(y;\xi_0)}.$$

Based on the comparison of the BLAST and the sequential analysis procedures, it can be argued that a suitable score to use in BLAST should also be the logarithm of a likelihood ratio. Under this argument, if the amino acid pair (j,k) is observed at any position, and if $p_j p'_k$ and $q(j,k)$ are, respectively, the null and the alternative hypothesis probabilities of this pair, the (discrete random variable) score $S(j,k)$ becomes

$$S(j,k) = \log \frac{q(j,k)}{p_j p'_k}. \tag{10.6}$$

Any score proportional to $S(j,k)$ is also reasonable.

The second argument favoring the choice (10.6) for the score associated with the pair (j,k) is more subtle (Karlin and Altschul (1990)). This argument also leads to the choice of a specific proportionality constant. Suppose some arbitrary substitution matrix is chosen, with (j,k) element $S(j,k)$. Now let $q(j,k)$ be defined implicitly by

$$S(j,k) = \lambda^{-1} \log \frac{q(j,k)}{p_j p'_k}, \tag{10.7}$$

where λ is defined in equation (10.3), and thus explicitly by

$$q(j,k) = p_j p'_k e^{\lambda S(j,k)}. \tag{10.8}$$

The right-hand side is the typical term on the left hand-side in equation (10.3). Therefore $\sum_{j,k} q(j,k) = 1$. Thus the $q(j,k)$ (which are all positive) form a probability distribution. This is not an arbitrary distribution. Karlin and Altschul (1990) and Karlin (1994) show that in practice, when the null hypothesis is true, the frequency with which the observation (j,k) arises in high-scoring excursions, where the score used is as given in equation (10.7), is asymptotically equal to $q(j,k)$. They then argue that a scoring scheme is "optimal" if the frequency of the observation (j,k) in high-scoring excursions is asymptotically equal to the "target" frequency $q(j,k)$, the frequency arising if the alternative hypothesis is true, (i.e., the frequency in the most biologically relevant alignments of conserved regions). This, then, argues for the use of $S(j,k)$ as defined in equation (10.7) as the score statistic.

These arguments lead us to adopt the following procedure. Suppose that the alternative hypothesis specifies a well-defined probability $q(j,k)$ for the amino acid pair (j,k), while the null hypothesis specifies a probability $p_j p'_k$ for this pair. Then we define the score $S(j,k)$ associated with this pair as that given by equation (10.7).

These arguments do not yet specify how to determine the most appropriate form for the $q(j,k)$'s. There are various possibilities for this. One frequently adopted choice is that deriving from the evolutionary arguments that lead to the PAMn matrix construction described in Section 6.5.3. In the notation of Section 6.5.3,

$$q(j,k) = p_j m_{jk}^{(n)}, \tag{10.9}$$

so that

$$S(j,k) = \log \frac{m_{jk}^{(n)}}{p'_k}. \tag{10.10}$$

The values of the $q(j,k)$'s for the simple symmetric model of Section 6.5.3 are given in equation (6.35) for one specific value of n. The derivation of these values emphasizes that $q(j,k)$ is a function of n, and that some extrinsic choice of a reasonable value of n must be made to use PAMn matrices in BLAST methods. We discuss aspects of this choice in Section 10.6.

The choice of $S(j,k)$ as the logarithm of a likelihood ratio can be related to the concepts of relative entropy and support discussed in Section 1.14.2. Specifically, the score defined by equation (10.7) is proportional to the support given by the observation (j,k) in favor of the alternative hypothesis over the null hypothesis. Equation (1.124) shows that when the alternative hypothesis is true, the mean H of this support is

$$H = \sum_{j,k} q(j,k) \log \frac{q(j,k)}{p_j p'_k}, \tag{10.11}$$

and this is the relative entropy defined in equation (1.119). Equation (10.7) shows that this relative entropy can be written as

$$H = \sum_{j,k} q(j,k) \lambda S(j,k) = \lambda E\left(S(j,k)\right), \tag{10.12}$$

the expected value being taken assuming that the alternative hypothesis is true.

From the discussion following (10.8), if the score $S(j,k)$ for the pair (j,k) is defined as in (10.7), the mean score in high-scoring segments is asymptotically $\sum_{j,k} q(j,k)S(j,k)$, and from (10.12) this is

$$\lambda^{-1}H. \tag{10.13}$$

This asymptotic result is used in BLAST calculations (see Section 10.3.3).

Simulations, however, show that the convergence to this asymptotic value is very slow. For the symmetric PAMn substitution matrix discussed in Section 6.5.4 with $n = 259$, and with equal amino acid frequencies, the asymptotic value $\lambda^{-1}H$ of the mean step size in high-scoring segments, found from computation of λ and H from (10.3) and (10.11), respectively, is 0.446. This is identical to the value given in equation (6.36), found from Markov chain considerations. For this example, Table 10.1 shows simulation estimates of this mean for various values of N, the length of the alignment. The slow rate of convergence to the asymptotic value 0.446 is clear. This observation will be relevant to the edge correction formula discussed in Section 10.3.3.

N	500	5,000	50,000	500,000	5,000,000	limiting value
mean step	1.021	.712	.608	.560	.533	.446

Table 10.1. Simulation values for the mean step size in maximally-scoring segments, as a function of N. Simulations performed with 10,000 to 1,000,000 repetitions.

The value of the relative entropy H appears on BLAST printouts. However, the calculation used for these printouts is slightly different from that implied by (10.12). The value of $q(j,k)$ used to compute the score $S(j,k)$ may well be unknown, so that while the values of λ and $S(j,k)$ are known, direct computation of H as defined in (10.12) is not possible.

The BLAST printout value of H uses an indirect approach. With the values of λ, the $S(j,k)$, and the p_j and p'_k in hand, $q(j,k)$ is calculated by using equation (10.8). The printout value of H is now calculated as in (10.12), using the values of $q(j,k)$ so calculated.

10.2.5 Bounds and Approximations for the BLAST P-Value

We have seen that the test statistic used in BLAST is the maximum $Y_{\max}$ of $n \cong N/A$ random variables, each being a random upwards excursion height following a ladder point in the BLAST random walk. The theory of Section 7.6.4 shows that each upward excursion has, approximately and asymptotically, the geometric-like distribution (1.19). We use this result in this section to obtain asymptotic bounds for the null hypothesis distribution of $Y_{\max}$ and hence asymptotic bounds for a BLAST P-value. An approximation used in some BLAST implementations for this P-value will also be given.

The analysis of Section 2.11.3 shows that there exists an asymptotic distribution for the maximum of n iid continuous random variables whose density function has support of the form $(A, +\infty)$. The BLAST test statistic $Y_{\max}$ is, however, a discrete random variable, and an asymptotic distribution for the maximum of n iid discrete random variables, analogous to that for continuous random variables, is known not to exist. On the other hand it is possible to use the continuous distribution results to find asymptotic bounds for the distribution of $Y_{\max}$. The procedure is as follows.

If $X_{\max}$ is the maximum of n iid continuous random variables, and if $Y_{\max} = \lfloor X_{\max} \rfloor$ is the integer part of $X_{\max}$, then $Y_{\max}$ is a discrete random variable and

$$X_{\max} - 1 < Y_{\max} \leq X_{\max}.$$

Thus for any positive integer y,

$$\text{Prob}(X_{\max} \leq y) \leq \text{Prob}(Y_{\max} \leq y) \leq \text{Prob}(X_{\max} \leq y + 1). \tag{10.14}$$

Let $X_{\max}$ be the maximum of n iid random variables each having the exponential distribution (1.66), and put $Y_{\max} = \lfloor X_{\max} \rfloor$. Then the argument surrounding equations (2.115) and (2.116) shows that $Y_{\max}$ has the same distribution as the maximum of n iid random variables, each having the geometric distribution (1.69). Application of (2.130) and the bounds in (10.14) shows that to a close approximation,

$$e^{-ne^{-\lambda y}} \leq \text{Prob}(Y_{\max} \leq y) \leq e^{-ne^{-\lambda(y+1)}}, \tag{10.15}$$

or equivalently

$$1 - e^{-ne^{-\lambda y}} \leq \text{Prob}(Y_{\max} \geq y) \leq 1 - e^{-ne^{-\lambda(y-1)}}, \tag{10.16}$$

for any positive integer y.

This discussion suggests how a parallel calculation for the maximum of random variables having a geometric-like distribution can be obtained. If $Y_{\max}$ is the maximum of n iid random variables, each having the geometric-like distribution given in (1.74), then a calculation analogous to that leading to (10.16) gives the approximate asymptotic inequality

$$1 - e^{-nCe^{-\lambda y}} \leq \text{Prob}(Y_{\max} \geq y) \leq 1 - e^{-nC\,e^{-\lambda(y-1)}}. \tag{10.17}$$

Whereas for the geometric distribution the upper bound in (10.15) and the lower bound in (10.16) hold even for small y, the inequalities in (10.17) are ultimately based on the *asymptotic* expression (1.74), which applies for large values of y as well as large n. They might not hold for small values of y, even when n is large.

If we now replace n by N/A for the mean number of BLAST ladder points and define a new parameter K by

$$K = \frac{C}{A} e^{-\lambda}, \tag{10.18}$$

the inequality (10.17) becomes

$$1 - e^{-NKe^{-\lambda(y-1)}} \leq \text{Prob}(Y_{\max} \geq y) \leq 1 - e^{-NK\, e^{-\lambda(y-2)}}. \tag{10.19}$$

If we replace y in this inequality by $x + \lambda^{-1} \log N$, we obtain

$$e^{-Ke^{-\lambda(x-1)}} \leq \text{Prob}(Y_{\max} - \lambda^{-1} \log N \leq x) \leq e^{-Ke^{-\lambda x}}, \tag{10.20}$$

or equivalently

$$1 - e^{-Ke^{-\lambda x}} \leq \text{Prob}(Y_{\max} \geq \lambda^{-1} \log N + x) \leq 1 - e^{-Ke^{-\lambda(x-1)}}. \tag{10.21}$$

Allowing for notational changes, this is identical to one of the inequalities (1.13) in Karlin and Dembo (1992). Equivalently, for any value $y_{\max}$,

$$1 - e^{-KNe^{-\lambda y_{\max}}} \leq \text{Prob}(Y_{\max} \geq y_{\max}) \leq 1 - e^{-KNe^{-\lambda(y_{\max}-1)}}. \tag{10.22}$$

These inequalities give bounds for the P-value corresponding to any observed value $y_{\max}$ of $Y_{\max}$. These bounds for a BLAST P-value are not directly relevant in practice, since in practice a BLAST search involves the comparison of a short query sequence with a large database, consisting of many fragments, and there is no a priori alignment of the query sequence with any part of the database. This fact introduces various complications which we shall take up in the following sections. Nevertheless, we shall see that the P-value approximation used in the implementation of BLAST described in Section 10.5 derives ultimately from the lower P-value bound in (10.22).

It is often difficult to calculate P-values even for relatively simple random variables, so it is remarkable that P-values can be approximated with the comparatively simple and efficient procedure described above. On the other hand, we shall see in Section 11.6.1 that while the approximation is often conservative (that is, can overestimate the true P-value), it can also be anticonservative, that is underestimate it. This might be because the geometric-like distribution on which the bounds in (10.22) are ultimately based is an asymptotic one, and might not apply for comparatively small values of $y_{\max}$. Also, it would be more appropriate to use the conservative upper bound in (10.22) rather than the lower bound.

The calculation of the bounds in (10.22) requires calculation of both λ and K. The computation of λ via equation (10.3) is comparatively straightforward. The calculation of K from the right-hand side in equation (10.18) would require the calculation of C, A, and λ. However, an exact calculation of K is straightforward in at least two cases. The first of these arises when the largest of the $S(j, k)$ is $+1$, and the second when the smallest $S(j, k)$

is -1, arising for example in the simple DNA scoring scheme described in Section 7.1. Using the notation S for the size of a step in the BLAST random walk, the two respective formulae for K are

$$K = \left(e^{-\lambda} - e^{-2\lambda}\right) E\left(Se^{\lambda S}\right) \tag{10.23}$$

and

$$K = \left(e^{-\lambda} - e^{-2\lambda}\right) \frac{(E(S))^2}{E\left(Se^{\lambda S}\right)}, \tag{10.24}$$

the expectations being taken assuming that the null hypothesis (10.1) is true.

10.2.6 The Normalized and the Bit Scores

Karlin and Altschul (1993) call the expression

$$\lambda Y_{\max} - \log(NK) \tag{10.25}$$

a "normalized score," denoted here by S'. In terms of this score, the inequalities (10.20) can be written as

$$e^{-e^{\lambda} e^{-s}} \leq \operatorname{Prob}(S' \leq s) \leq e^{-e^{-s}}. \tag{10.26}$$

From the upper inequality we obtain the approximation

$$\operatorname{Prob}(S' \geq s) \cong 1 - e^{-e^{-s}}. \tag{10.27}$$

The P-value corresponding to an observed value $s' = \lambda y_{\max} - \log(NK)$ of S' is, from (10.27),

$$P\text{-value} \cong 1 - e^{-e^{-s'}}. \tag{10.28}$$

This is identical to the approximation given by the lower bound in (10.22). When s is large, (10.27) may be further approximated by

$$\operatorname{Prob}(S' \geq s) \cong e^{-s}. \tag{10.29}$$

The similarity between the approximation (10.27) and equation (2.127) is of course no coincidence, since S' is a (normalized) extreme value. The approximation (10.27) and the fact that the mean and variance of the density function whose cumulative distribution function is (2.126) are respectively γ (Euler's constant) and $\pi^2/6$ show that these are approximately mean and variance of S'. From (10.25) and the linearity property of a mean (see Section 1.4), the approximate null hypothesis mean and variance of $Y_{\max}$ are, approximately,

$$\lambda^{-1}\left(\log(KN) + \gamma\right), \quad \text{and} \quad \pi^2/(6\lambda^2) \tag{10.30}$$

respectively. The value of γ is usually much smaller than KN, and in BLAST calculations the mean is often approximated by

$$\lambda^{-1}\left(\log(KN)\right). \tag{10.31}$$

BLAST printouts or published papers record a score similar to the normalized score S', namely the "bit" score. In more recent printouts this is defined by

$$\text{bit score} = \frac{\lambda Y_{\max} - \log K}{\log 2}. \tag{10.32}$$

Previous printouts recorded a bit score defined by

$$\text{bit score} = \frac{\lambda Y_{\max}}{\log 2}. \tag{10.33}$$

The bit score (10.33) has an invariance property, since its value, and hence its probability distribution, does not change if all entries in the substitution matrix used are multiplied by the same constant, say G. This can be seen from the fact that such a multiplication changes the value of $Y_{\max}$ by a multiplicative factor of G, but at the same time (see equation (10.3)) changes the value of λ by a multiplicative factor $1/G$. Thus the bit score (10.33) remains unchanged by this multiplication.

If the expression on the right-hand side of (10.27) is used to approximate the distribution of the normalized score S', then to this level of approximation the normalized score has a distribution also having the invariance property, since the right-hand side in (10.27) is free of any parameter.

A much stronger result than this is true. Whereas the value of $Y_{\max}$ has no absolute interpretation if the substitution matrix from which it is calculated is not specified, the normalized score S' and the bit score do have such an interpretation. In the case of S' this is made clear by approximations such as (10.27): Here the right-hand side is free of any parameters, so that the (approximate) distribution of S' is known whatever the details of the substitution matrix. If N is given, the same can be said for the bit score (10.32).

10.2.7 The Number of High-Scoring Excursions

In this section we define and discuss the quantity E', whose calculation leads ultimately to the quantity "Expect" found on BLAST printouts. Throughout the discussion we ignore edge effects: These are discussed in detail in Section 10.3.3.

Consider excursions from a ladder point in the random walk described by the comparison of the two sequences. We have seen that under the null hypothesis, for each such excursion, the maximum height Y has a geometric-like distribution whose parameters can be calculated. Denoting as above the maximum heights of the excursions from the various ladder points by $Y_1, Y_2, \ldots$, the relation (1.74) shows that the probability that any Y_i takes a value v or larger is approximately $Ce^{-\lambda v}$, where C and λ are those appropriate to the walk in question. Since to a close approximation the number of excursions can be taken to be N/A, as discussed in Section 10.2.3, the mean number of excursions reaching a height v or more is approximately $\frac{NC}{A}e^{-\lambda v}$. In the standard BLAST calculations discussed in the printout in Section 10.5, this mean is replaced by the approximating value

$$NKe^{-\lambda v}, \tag{10.34}$$

where K is given by (10.18). In Section 10.4 we shall trace back the printout P-value calculation to the expected value expression (10.34).

Since $Y_1, Y_2, \ldots$ are iid random variables, the number of excursions having a height v or more has a binomial distribution with mean given by (10.34). The theory developed in Section 4.2 shows that when v is large, the number of excursions reaching a height greater than or equal to v, that is, the number of high-scoring segment pairs (HSPs) with a score v or more, has, using the Poisson approximation to the binomial, a Poisson distribution with mean given in (10.34). (A more sophisticated analysis, based on equations such as (2.81) that allow for the fact that the number of ladder points is a random variable, arrives at the same conclusion.) Thus, to test for significance, the actual number of such excursions achieving a score exceeding v can be compared with the tail probability of this Poisson distribution.

The expected value of the number of excursions corresponding to the observed maximal score $y_{\max}$ is found by replacing the arbitrary number v in equation (10.34) by $y_{\max}$. This expected value is denoted by E', so that

$$E' = NKe^{-\lambda y_{\max}}. \tag{10.35}$$

The relation between E' and the normalized score S' defined in (10.25) is

$$S' = -\log E', \tag{10.36}$$

and the relation between E' and the P-value approximation is found from (10.28) as

$$P\text{-value} \cong 1 - e^{-E'}, \quad E' = -\log(1 - P\text{-value}). \tag{10.37}$$

It follows from the approximation (B.21) that the approximate P-value is very close to E' when E' is small.

Similar calculations may be made for any high-scoring excursion.

10.2.8 The Karlin–Altschul Sum Statistic

Focusing on the value of $Y_{\max}$ loses the information provided by the heights of the second-largest, third-largest, etc., excursions in the random walk. In this section we discuss a statistic that uses information from these other excursions.

Consider the r largest excursion heights, that is, the r largest Y_i values, assuming that there are at least r ladder points. It is convenient to use a notation that is different from the notation for order statistics used in Chapter 2, and assume that $Y_1 (= Y_{\max}) \geq Y_2 \geq \cdots \geq Y_r$. By analogy with the definition in equation (10.25) we can compute r normalized scores $S'_1, S'_2, \ldots, S'_r$ from Y_1, $Y_2, \ldots$, Y_r, where

$$S'_i = \lambda Y_i - \log(NK). \tag{10.38}$$

Note that $S'_1 = S'$ as defined in equation (10.25).

Karlin and Altschul (1993) show that to a close approximation, the null hypothesis joint density function $f_{\boldsymbol{S}}(s_1, \ldots, s_r)$ of $\boldsymbol{S} = (S'_1, \ldots, S'_r)$ is

$$f_{\boldsymbol{S}}(s_1, \ldots, s_r) = \exp\left(-e^{-s_r} - \sum_{k=1}^{r} s_k\right). \tag{10.39}$$

We can use any reasonable function of $S'_1, S'_2, \ldots, S'_r$ as test statistic. Transformation methods such as those introduced in Chapter 2 can then be used to find the distribution of this test statistic, and this in turn allows the computation of a P-value, and also an E, or Expect, value, corresponding to any observed value of this statistic.

The specific statistic suggested by Karlin and Altschul (1993) is the sum $T_r = S'_1 + \cdots + S'_r$ of the normalized scores. This is called the Karlin–Altschul sum statistic. Using transformation methods such as those described in Section 2.13, Karlin and Altschul use the joint density function (10.39) to calculate the null hypothesis density function $f(t)$ of T_r. The resulting expression is

$$f_{T_r}(t) = \frac{e^{-t}}{r!(r-2)!} \int_0^{+\infty} y^{(r-2)} \exp(-e^{(y-t)/r}) dy. \tag{10.40}$$

As an exercise in transformation theory we confirm this calculation for the case $r = 2$ in Appendix D. When t is sufficiently large, this density function can be used to find the approximate expression

$$\text{Prob}(T_r \geq t) \cong \frac{e^{-t} t^{r-1}}{r!(r-1)!}. \tag{10.41}$$

In the case $r = 1$, this is the approximation given in equation (10.29). The approximation (10.41) is sufficiently accurate when $t > r(r + 1)$, and popular implementations of BLAST use (10.41) when this inequality holds.

If t is the observed value of T_r, the right-hand side in (10.41) then provides the approximate P-value corresponding to this observed value. This is used as a component of the eventual BLAST printout P-value.

Karlin and Altschul (1993) provide an example (see their Table 1) in which the observed values of the highest two normalized scores are $s'_1 = 4.4$ and $s'_2 = 2.5$. Using the value $r = 1$ in the approximation (10.41), the P-value corresponding to the highest normalized score 4.4 is $e^{-4.4} = 0.012$. Using the value $r = 2$, the P-value corresponding to the sum 6.9 of the highest two normalized scores is calculated from (10.41) as $\frac{6.9}{2} e^{-6.9} = 0.0035$, and these calculations confirm those given by Karlin and Altschul. For further aspects of these calculations, and of the calculations in their Table 2, see Problems 10.13 and 10.14.

A further aspect of the use of a test statistic based on T_r is that of consistent ordering. We say that r HSPs, HSP1, HSP2, ..., HSPr, between two sequences are *consistently ordered* if whenever the midpoint in the first sequence in HSPi comes before the midpoint in the first sequence in HSPj, then the same is true for the midpoints of the second sequence. More generally, one might require that the sequences in the different HSPs not overlap, or overlap no more than some fixed proportion (in popular implementations of BLAST, the default value of this proportion is 0.125). When consistent ordering is required, the P-value calculations must be amended. In the case where overlaps are unrestricted, this requirement cuts down the search space by a factor of $r!$, the number of orderings of the r HSPs. This implies that the P-value calculated from (10.41) should be divided by $r!$. A simple approximation (Karlin and Altschul (1993)) is that P-value calculations are amended by replacing t, the observed value of T_r, by $t + \log(r!)$ in the right-hand side of (10.41), or by a corresponding amendment to the calculations using (10.40). The popular implementations of BLAST use this approach, and furthermore allow the degree to which the HSPs overlap to be restricted. Restricting overlaps should require a further adjustment of the P-value. This is not apparently done by the popular implementations. However, an adjustment is made to the "edge correction" factor discussed below, which may or may not account for this (see Section 10.3.3).

A further complication introduced by the use of the sum statistic in BLAST is that of multiple testing. In practice, the value of r is not fixed in advance and is allowed to vary. Thus the problem of multiple testing, discussed in Section 3.11, arises. We delay discussion of the way in which this problem is handled in BLAST calculations until Section 10.3.4.

In BLAST printouts the notation r is replaced by N. We have used r here because N is used to denote a sequence length. Further, r is the notation used in the fundamental paper of Karlin and Altschul (1993).

10.3 The Comparison of Two Unaligned Sequences

10.3.1 Introduction

The theory of Section 10.2 considered calculations relevant to a fixed ungapped alignment in the comparison of two sequences each of length N. In this section we consider a more general question. We are given two sequences of lengths N_1 and N_2, but we are not given any specific alignment between them. The goal is to find the significance of high-scoring segment pairs between all possible (ungapped) local alignments. The highest-scoring pair is called the maximal-scoring segment pair (MSP).

10.3.2 Theoretical and Empirical Background

BLAST considers all ungapped alignments determined by all possible relative positions of the two sequences. For each relative position, the alignment is extended as far as possible in either direction, giving a total of N_1+N_2-1 ungapped alignments. Figure 10.1 shows the first five alignments between two sequences of length 11 and 9 respectively.

```
sequence 1    . . . . . . . . . . .
sequence 2                        . . . . . . . . .

sequence 1    . . . . . . . . . . .
sequence 2                      . . . . . . . . .

sequence 1    . . . . . . . . . . .
sequence 2                    . . . . . . . . .

sequence 1    . . . . . . . . . . .
sequence 2                  . . . . . . . . .

sequence 1    . . . . . . . . . . .
sequence 2                . . . . . . . . .
```

Figure 10.1.

Each such alignment yields a random walk similar to that considered in Section 10.2.2, giving a collection of random walks. There are N_1N_2 amino acid comparisons that can be made as the two sequences take all possible positions relative to each other.

The theory for this case is far more complicated than that outlined in Section 10.2, where only one alignment occurs. Among other matters the question of the dependence of the walks arising in different alignments must be addressed. The key papers developing the theory are Dembo et al. (1994a, 1994b). The theory is too advanced for this book, and here we simply reproduce the relevant results, the most important of which is that, to a sufficient approximation, many of the conclusions of Section 10.2 carry over to the present case, with N replaced by N_1N_2.

However, there are several qualifications to make about this statement. First, several conditions (given by Dembo et al. (1994a, 1994b)) need to be satisfied before the theory of Section 10.2 can be used. Second, some of the theoretical results proved apply only in the limit as both N_1 and N_2 become large. Thus the theory might not hold in the case of interest in practice, where both sequences might be of length only a few hundred or less. Thus many simulations have been carried out to assess the extent to which the theory of Section 10.2 carries through to cases of practical interest; see, for example, Altschul and Gish (1996) and Pearson (1998). A broad conclusion reached from these simulations is that the theory of Section 10.2 does carry over to a reasonable approximation if N is replaced by the product N_1N_2, or by a more refined function allowing for edge effects. Thus with much but not complete theoretical and empirical support, and remembering that cases can arise that are not covered by the theory of Section 10.2, we now use that theory for the comparison of two sequences, replacing N by N_1N_2 for the moment, and by a more refined expression in Section 10.3.3.

We consider first the random variable $Y_{\max}$, the maximum score achieved in the random walk comparing the sequences, using all possible ungapped local alignments between the two. This score corresponds to the MSP. Any MSP or HSP starts at a ladder point in the BLAST random walk and finishes the first time that the maximum upward excursion from this ladder point is reached. Under the heuristic adopted, $Y_{\max}$ is the maximum of a number of geometric-like random variables, whose distribution depends on the parameters λ, C, and n. The calculations for λ and C follow as in Section 10.2.3. The mean number of ladder points in this random walk corresponding to the collection of all alignments of the two sequences is approximated by

$$\frac{N_1N_2}{A}, \tag{10.42}$$

where A is the mean distance between ladder points. This value is used throughout the following theory. The discussion at the end of Section 10.2.3 applies equally well to explain this formula. The theory discussed in Section 10.2.3 can now be used, with the value given by equation (10.42) for the mean number of ladder points, the value of C given by equation (7.61), and the value of λ given by equation (10.3).

The key formulae discussed above are now taken over to the present case with these parameter values. Thus assuming that the null hypothesis (10.1) is true, the inequalities (10.21) are replaced by

$$1-e^{-Ke^{-\lambda x}} \leq \text{Prob}(Y_{\max} > \lambda^{-1}\log(N_1N_2)+x) \leq 1-e^{-Ke^{-\lambda(x-1)}}, \tag{10.43}$$

and if the normalized score S' is redefined as

$$S' = \lambda Y_{\max} - \log(N_1N_2K), \tag{10.44}$$

the inequality (10.26) and the approximations (10.27) and (10.29) continue to hold. As a result, the right-hand side in the latter approximation also has the interpretation of an approximate P-value corresponding to the observed value s of S' as defined in (10.44).

Similarly, the expected number E' of excursions reaching a height $y_{\max}$ or more is found by replacing equation (10.35) by

$$E' = N_1N_2Ke^{-\lambda y_{\max}}, \tag{10.45}$$

and the approximate null hypothesis mean of $Y_{\max}$ is

$$\lambda^{-1}\left(\log(N_1N_2K) + \gamma\right). \tag{10.46}$$

10.3.3 Edge Effects

The calculations of the preceding sections do not allow for edge effects, an important factor in the comparison of two comparatively short sequences. In this section we discuss the adjustments to the previous calculations that are used in BLAST calculations to allow for edge effects.

A high-scoring random walk excursion induced by the comparison of the two sequences might be cut short at the end of a sequence match, so that the height of high-scoring excursions, and the number of such excursions, will tend to be less than that predicted by the theory above. Whereas much of BLAST theory concerns two long sequences for which edge effects are of less importance, in practice BLAST considers databases made up of a large number of often short sequences, for which edge effects are important. Thus BLAST calculations allow for edge effects, and do this by subtracting from both N_1 and N_2 a factor depending on the mean length of any high-scoring excursion. The justification for this is largely empirical (Altschul and Gish (1996)).

Equation (10.13) shows that the mean value of the step in a high-scoring excursion asymptotically approaches the value $\lambda^{-1}H$. Given that the height achieved by a high-scoring excursion is denoted by y, equation (7.23) suggests that the mean length $E(L|y)$ of this excursion, conditional on y, is given by

$$E(L|y) = \frac{\lambda y}{H}. \tag{10.47}$$

BLAST theory then replaces N_1 and N_2 in the calculations given above respectively by $N_1' = N_1 - E(L)$, $N_2' = N_2 - E(L)$. Specifically, the normalized score (10.25) is replaced by

$$\lambda Y_{\max} - \log(N_1' N_2' K), \tag{10.48}$$

with

$$N_1' = N_1 - \frac{\lambda Y_{\max}}{H}, \quad N_2' = N_2 - \frac{\lambda Y_{\max}}{H}. \tag{10.49}$$

The expression (10.34) for the expected number of excursions scoring v or higher is correspondingly replaced by

$$N_1' N_2' K e^{-\lambda v}, \tag{10.50}$$

with $N_1' = N_1 - \lambda v/H$, $N_2' = N_2 - \lambda v/H$. Similarly, the calculation of E' given in (10.35) is replaced by

$$E' = N_1' N_2' K e^{-\lambda y_{\max}}. \tag{10.51}$$

The use of edge corrections using (10.49) assumes that the asymptotic formula (10.13) for the mean step size in a high-scoring excursion is appropriate. The simulations discussed in Section 10.2.4 show that this might not be the case, at least when N_1 and N_2 are both of order 10^2. Table

10.2 shows empirical MSP mean lengths (from simulations with 10,000 to 1,000,000 replications) and the values calculated from (10.47) for the simulation leading to the data of Table 10.1. Clearly the values calculated from (10.47) are inaccurate for anything other than very large values of N. Thus while the calculated values approach the empirical values as N increases (in line with the convergence of the mean step sizes to the asymptotic value in Table 10.1), the use of the edge correction implied by (10.49) might in practice lead to P-value estimates less than the correct values, that is, to anti-conservative tests, for anything other than very large values of N. The use of the observed value of the length of the MSP appears to give more accurate results (Altschul and Gish (1996)).

N	500	5,000	50,000	500,000	5,000,000
Empirical mean length	43.2	106.8	181.1	258.0	335.9
Calculated mean length	98.7	168.4	237.4	301.8	373.9

Table 10.2. Empirical values for the mean length of the MSP and the value found from (10.47) and empirical values of $y_{\max}$. Simulations performed with 10,000 to 1,000,000 repetitions.

In the popular implementations of BLAST the edge effect correction factor for the Karlin–Altschul sum statistic T_r is calculated as follows. First, a raw edge effect correction is calculated as $\lambda(Y_1 + Y_2 + \cdots + Y_r)/H$, generalizing the term $\lambda Y_{\max}/H$ given in (10.49). When consistent ordering is required and overlaps are restricted by a factor f, this is then multiplied by a factor $1 - (r+1)f/r$, where f is an "overlap adjustment factor" that can be chosen by the investigator. The default value of f is 0.125, implying that overlaps between segments of up to 12.5% are allowed. The use of f is illustrated by an example in Section 10.5.2. To this the value $r - 1$ is added, leading eventually to an edge correction value $E(L)$, defined by

$$E(L) = \frac{\lambda}{H}(Y_1 + Y_2 + \cdots + Y_r)\left(1 - \frac{r+1}{r}f\right) + r - 1. \tag{10.52}$$

While this formula is used in BLAST, there appears to be no publication justifying its validity. It could be tested empirically in the spirit of Altschul et al. (1996). The values of N_1 and N_2 in the normalized score formula (10.38) are then replaced, respectively, by

$$N_1' = N_1 - E(L), \; N_2' = N_2 - E(L). \tag{10.53}$$

The normalized scores in (10.38) are now redefined as

$$S_i' = \lambda Y_i - \log(N_1' N_2' K), \tag{10.54}$$

and with this new definition the sum statistic T_r is redefined as

$$T_r = S_1' + S_2' + \cdots + S_r'. \tag{10.55}$$

The problems discussed above concerning the accuracy of the approximation (10.47) leading to the expressions for N_1' and N_2', and hence of calculations derived from S_i' and T_r, apply here also.

If the r HSPs are required to be consistently ordered, a term $\log r!$ is added to T_r (as discussed in Section 10.2.8), and if the sum so calculated is t, the P-value is then calculated as in (10.41).

10.3.4 Multiple Testing

There is no obvious choice for the value of r when the sum statistic is used in the test procedure. It is natural to consider all $r = 1, 2, 3, \ldots$, and choose the set of HSPs with lowest sum statistic P-value as the most significant, regardless of the value of r, and this is what is done in BLAST calculations. However, this procedure implies that a sequence of tests, one for each r, rather than a single test, is performed, so that the issue of multiple testing, discussed in Section 3.11, arises. Green (unpublished results) has found through simulations that ignoring the multiple testing issue leads to a significant overestimate of BLAST P-values, so that an amendment to formal P-value calculations is indeed necessary.

Unfortunately, there is no rigorous theory available to deal with this issue, and in practice it is handled in an ad hoc manner. For example, in the Washington University versions of BLAST, the P-value is adjusted when $r > 1$ by dividing the formal P-value by a factor of $(1-\pi)\pi^{r-1}$. The parameter π has default value .5, but its value can be chosen by the user. The default value 0.5 is used in the example in Section 10.5.

When $r = 1$ the procedure is slightly different. The factor $(1-\pi)\pi^{r-1}$ in this case is $1-\pi$, and this implies that the value of E' given in (10.35) is divided by $1-\pi$ to find the amended expected value E. The BLAST default value 0.5 of π implies that $E = 2E'$, so that E is calculated to be

$$E = 2N_1'N_2'Ke^{-\lambda y_{\max}}. \tag{10.56}$$

The P-value corresponding to this is then found, using the analogy with (10.37), to be

$$P\text{-value} \cong 1 - e^{-E}. \tag{10.57}$$

The P-value and Expect calculations used in BLAST embody the amendments discussed above to the theoretical values (given in Section 10.2). These amendments relate to edge effect corrections, multiple testing corrections and, in the case of the sum statistic, the consistent ordering and overlap corrections. Some details of these amendments appear not to be mentioned in BLAST documentation in the popular implementations of BLAST, and only become clear by careful reading of the code.

10.4 The Comparison of a Query Sequence Against a Database

We now consider the case that is most relevant in practice. In this case we have a single “query” sequence, and we wish to search an entire database of many sequences for those with significant similarity to the query sequence. To do this, first a (heuristic) search algorithm is employed to find the high-scoring HSPs, or sets of HSPs. The P-values and Expect values of these HSPs are then approximated. These approximations are discussed in this section.

Whereas query sequence amino acid frequencies are taken from the query sequence at hand, database frequencies often are taken from some (different) published set of estimated amino acid frequencies, for example those in Robinson and Robinson (1991). These might be different again from those used to create the substitution matrix.

In approximating database P-values and Expect values, the size of the entire database must be taken into account. This raises another multiple testing problem in addition to that discussed above. What is done in practice is first to use the results of the last section to compare the query sequence to each individual database sequence, to obtain P-values for individual sequence comparisons. Then the individual sequence P-values are adjusted to account for the size of the database. If all the sequences in the database were the same size, then we could just multiply the Expect values by the number of sequences, using the linearity property of means (see (2.66)). As an approximation to this, what is done in practice is to multiply by D/N_2, where D is the total length of the database, (i.e., the sum of the lengths of all of the database sequences), and N_2 is the length of the database sequence that aligns with the query to give the HSP (or HSPs) in question. These Expect values are then converted to P-values. The details are as follows.

We consider first the case of single HSPs ($r = 1$). Because of its linearity properties, the most useful quantity for database searches is the quantity E, defined in (10.56), and its generalizations for other HSPs. Suppose that in the database sequence of interest there is some HSP with score v. The Poisson distribution is then used to approximate the probability that in the match between query sequence and database sequence at least one HSP scores v or more. This probability is approximately

$$1 - e^{-E}. \tag{10.58}$$

Since the entire database is D/N_2 times longer than the database sequence of interest, the mean number of HSPs scoring v or more in the entire database, namely the BLAST printout quantity Expect, is given by

$$\text{Expect} = \frac{(1 - e^{-E})D}{N_2}. \tag{10.59}$$

From this value of Expect an approximate P-value is calculated from (10.37) as

$$P\text{-value} \cong 1 - e^{-\text{Expect}}. \tag{10.60}$$

This is the BLAST printout P-value for the case $r = 1$ in the implementation of BLAST discussed in Section 10.5

We shall see in Section 10.5 that the BLAST printout P-value is found by first using (10.59) to calculate "Expect" and then finding a P-value from (10.60). Once allowance is made for multiple testing, the size of the database, edge effects, and the multiple alignment situation, the value of "Expect" derives directly from the expression for E' – see the calculations in (10.62) and (10.63) – then back to (10.51) and thence to (10.34). Thus from the relation between a P-value and an expected value given in (10.37), the BLAST printout P-value traces back to the lower bound for a P-value given in (10.22), as was claimed below (10.34).

For the case $r > 1$, sum statistics for various database sequences are calculated as described in Section 10.2.8, and P-values are calculated either from (10.40) or (10.41), using all the amendments discussed above. From each such P-value a total database value of Expect is calculated using a formula generalizing that derived from (10.59), namely

$$\text{Expect} = \frac{(P\text{-value})D}{N_2}, \tag{10.61}$$

where N_2 is the length of the database sequence from which the sum is found. From this a P-value is calculated as in (10.60).

Finally, all single (i.e., $r = 1$) HSPs or summed ($r > 1$) HSPs with sufficiently low values of Expect (or, equivalently, sufficiently low P-values) are listed, and eventually printed out in increasing order of their Expect values. The value of r, given as N in the printout, is also listed.

10.5 Printouts

In this section we relate the above theory to an actual BLAST printout, describing the comparison of a query sequence with the Swiss Protein Database SWISS-PROT.

BLAST printouts give the values of λ, calculated from (10.3), of K, calculated from (10.18) amended appropriately for sequence comparisons, and H, found from the procedure described at the end of Section 10.2.4. They also list the statistics "Score" or "High Score," which in the case $r = 1$ are the values of the maximal scores $y_{\max}$ or other high-scoring HSPs. In the case of the sum statistic T_r (with $r > 1$), the score of the highest-scoring component in the sum is listed. Also listed are the "bit scores" associated with these, together with "Expect" values and P-values calculated as described in Section 10.4. We repeat that variants of these calculations are possible for different versions of BLAST and that sometimes more sophisticated calculations, taking into account factors not discussed above, are used.

10.5.1 Example

A partial printout of Example 3 from the Washington University BLAST 1.4 program follows:[1]

```
BLASTP 1.4.10MP-WashU [29-Apr-96] [Build 22:25:52 May 19 1996]


Query=  gi|557844|sp|P40582|YIV8_YEAST HYPOTHETICAL 26.8 KD PROTEIN IN HYR1
                               3'REGION.
        (234 letters)

Database:  SWISS-PROT Release 34.0
           59,021 sequences; 21,210,388 total letters.

----------------------------------------------------------------------
                                                                  Smallest
                                                                    Sum
                                                           High  Probability
Sequences producing High-scoring Segment Pairs:           Score  P(N)      N

sp|P46429|GTS2_MANSE GLUTATHIONE S-TRANSFERASE 2 (EC 2....   53  0.010     3
sp|P46420|GTH4_MAIZE GLUTATHIONE S-TRANSFERASE IV (EC 2...   70  0.14      1
sp|P41043|GTS2_DROME GLUTATHIONE S-TRANSFERASE 2 (EC 2....   54  0.19      2
sp|P34345|YK67_CAEEL HYPOTHETICAL 28.5 KD PROTEIN C29E4...   50  0.42      2
sp|Q04522|GTH_SILCU  GLUTATHIONE S-TRANSFERASE (EC 2.5....   62  0.87      1

----------------------------------------------------------------------

>sp|P46429|GTS2_MANSE GLUTATHIONE S-TRANSFERASE 2 (EC 2.5.1.18) (CLASS-SIG).
            Length = 203

 Score = 53 (24.4 bits), Expect = 0.010, Sum P(3) = 0.010
 Identities = 10/19 (52%), Positives = 15/19 (78%)

Query:   167 ISKNNGYLVDGKLSGADIL 185
             I+KNNG+L  G+L+ AD +
Sbjct:   136 ITKNNGFLALGRLTWADFV 154

 Score = 46 (21.2 bits), Expect = 0.010, Sum P(3) = 0.010
 Identities = 8/21 (38%), Positives = 13/21 (61%)

Query:    45 PELKKIHPLGRSPLLEVQDRE 65
             PE K   P G+ P+LE+  ++
Sbjct:    39 PEFKPNTPFGQMPVLEIDGKK 59
```

[1] http://sapiens.wustl.edu/blast/blast/example3-14.html

```
 Score = 36 (16.6 bits), Expect = 0.010, Sum P(3) = 0.010
 Identities = 8/26 (30%), Positives = 12/26 (46%)

Query:   202 EDYPAISKWLKTITSEESYAASKEKA 227
             E YP   K ++T+ S     A  + A
Sbjct:   173 EQYPIFKKPIETVLSNPKLKAYLDSA 198


>sp|P46420|GTH4_MAIZE GLUTATHIONE S-TRANSFERASE IV (EC 2.5.1.18) (GST-IV)
            (GST-27) (CLASS PHI).
            Length = 222

 Score = 70 (32.3 bits), Expect = 0.15, P = 0.14
 Identities = 17/56 (30%), Positives = 27/56 (48%)

Query:    18 RLLWLLDHLNLEYEIVPYKRDANFRAPPELKKIHPLGRSPLLEVQDRETGKKKILA 73
             R L  L+   ++YE+VP  R       PE    +P G+ P+LE  D    + + +A
Sbjct:    18 RALLALEEAGVDYELVPMSRQDGDHRRPEHLARNPFGKVPVLEDGDLTLFESRAIA 73


>sp|Q04522|GTH_SILCU GLUTATHIONE S-TRANSFERASE (EC 2.5.1.18) (CLASS-PHI).
            Length = 216

 Score = 62 (28.6 bits), Expect = 2.1, P = 0.87
 Identities = 15/43 (34%), Positives = 21/43 (48%)

Query:    18 RLLWLLDHLNLEYEIVPYKRDANFRAPPELKKIHPLGRSPLLE 60
             R+L  L   +LE+E VP    A     P    ++P G+ P LE
Sbjct:    15 RVLVALYEKHLEFEFVPIDMGAGGHKQPSYLALNPFGQVPALE 57

----------------------------------------------------------------------

Matrix name     Lambda    K       H
--------------------------------------
 BLOSUM62        0.320   0.137   0.401
```

The first calculation is to check the values of "Score" (equivalently "High score") in the printout, using the BLOSUM62 matrix in Table 6.7. As an example, the score 53 for the MANSE GLUTATHIONE match is calculated as $4 + 1 + \cdots + 1$, deriving from the $I - I$ match, the $S - T$ match, ..., the $L - V$ match in the first of the three components of the match of the query and the SWISS-PROT database. Other scores are found similarly.

We next verify the calculations leading to the Maize Glutathione match sequence value of 0.15 for Expect. For this case, the printout above shows that

$$N_1 = 234, \quad N_2 = 222, \quad y_{\max} = 70.$$

Equation (10.49), in conjunction with the printout values of λ and H, gives

$$N_1' = 234 - \frac{0.32(70)}{0.401} = 178, \quad N_2' = 222 - \frac{0.32(70)}{0.401} = 166.$$

Inserting these values and the printout value of K in (10.51), we get

$$E' \cong (178)(166)(0.137)e^{-0.32(70)} \cong 7.6(10)^{-7}. \tag{10.62}$$

Multiplying by the multiple testing factor 2 gives $E \cong 15.2(10)^{-7}$. Inserting this value in (10.59), we get

$$\text{Expect} \cong \left(1 - e^{-15.2(10)^{-7}}\right) \frac{21{,}210{,}388}{222} \cong 0.15, \tag{10.63}$$

in agreement with the value 0.15 for Expect found in the printout.

Given this value, equation (10.60) gives an approximate P-value of 0.14, in agreement with the printout calculation. Further, equation (10.33) gives a value $0.320(70)/\log 2 \cong 32.3$ for the bit score, in agreement with the printout value.

A similar set of calculations gives, to a close approximation, the value 2.1 for Expect in the Silcu Glutathione match.

We finally consider the Manse Glutathione match, for which $r = 3$, and describe the calculations leading to the printout value 0.010 for Expect. As noted above, this value of Expect is found using a series of amendment calculations, starting with the edge correction. The expression (10.52), together with data in the printout and the default value 0.125 for f, leads to an edge correction of

$$\frac{0.32}{0.401}(53 + 46 + 36)\left(1 - \frac{4}{3}(0.125)\right) + 2 = 91.78.$$

Thus from (10.53), $N_1' = 142.2835$ and $N_2' = 111.2835$. Using these values in (10.38), the amended observed value of T_3 is computed as

$$0.32(53 + 46 + 36) - 3\log\left((0.137)(142.22)(111.22)\right) = 20.16.$$

The consistent ordering requirement holds, so we add $\log 3! = 1.79$ to this to get the value 21.95. The P-value corresponding to this is found from (10.41) to be $1.181(10)^{-8}$. Multiplying by the multiple testing factor $2^3 = 8$ yields a value of $9.448(10)^{-8}$. The value of E is essentially identical to this.

Finally, the total database Expect value is found by multiplying this by 21,210,388/203, and this gives the value 0.010 found in the printout.

It might be a matter of concern that various somewhat arbitrary constants enter into the above calculations. This concern is reinforced by the fact that the cumulative distribution function of maximum statistics changes very sharply, as demonstrated in Table 3.4. As a result, calculated P-values are quite sensitive to the somewhat arbitrary numerical values of these constants. In practice, this concern is not important, since users of BLAST printouts seldom view a P-value even as small as 10^{-5} as interesting, and use the numerical P-values together with significant biological judgment.

10.5.2 A More Complicated Example

The way in which some BLAST outputs are formatted can be confusing. The partial output from a BLAST search against SWISS-PROT is given below,[2] in which only the set of HSPs between the query and one database sequence are shown. There are 12 HSPs in total; however, since consistent ordering is required, the smallest sum P-value comes from a set of 8 HSPs.

```
Query=  gi|604369|sp|P40692|MLH1_HUMAN MUTL PROTEIN HOMOLOG 1 (DNA MISMATCH
        (756 letters)

                                                                Smallest
                                                                  Sum
                                                         High  Probability
Sequences producing High-scoring Segment Pairs:         Score  P(N)      N

sp|P38920|MLH1_YEAST MUTL PROTEIN HOMOLOG 1 (DNA MIS...   675  1.7e-138  8

>sp|P38920|MLH1_YEAST MUTL PROTEIN HOMOLOG 1 (DNA MISMATCH REPAIR PROTEIN.)
            Length = 769

 Score = 675 (309.6 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 127/222 (57%), Positives = 170/222 (76%)

Query:   8 IRRLDETVVNRIAAGEVIQRPANAIKEMIENCLDAKSTSIQVIVKEGGLKLIQIQDNGTG 67
           I+ LD +VVN+IAAGE+I  P NA+KEM+EN +DA +T I ++VKEGG+K++QI DNG+G
Sbjct:   5 IKALDASVVNKIAAGEIIISPVNALKEMMENSIDANATMIDILVKEGGIKVLQITDNGSG 64

Query:  68 IRKEDLDIVCERFTTSKLQSFEDLASISTYGFRGEALASISHVAHVTITTKTADGKCAYR 127
           I K DL I+CERFTTSKLQ FEDL+ I TYGFRGEALASISHVA VT+TTK  + +CA+R
Sbjct:  65 INKADLPILCERFTTSKLQKFEDLSQIQTYGFRGEALASISHVARVTVTTKVKEDRCAWR 124

Query: 128 ASYSDGKLKAPPKPCAGNQGTQITVEDLFYNIATRRKALKNPSEEYGKILEVVGRYSVHN 187
            SY++GK+   PKP AG  GT I VEDLF+NI +R +AL++ ++EY KIL+VVGRY++H+
Sbjct: 125 VSYAEGKMLESPKPVAGKDGTTILVEDLFFNIPSRLRALRSHNDEYSKILDVVGRYAIHS 184

Query: 188 AGISFSVKKQGETVADVRTLPNASTVDNIRSIFGNAVSRELI 229
             I FS KK G++   +   P+ +  D IR++F  +V+  LI
Sbjct: 185 KDIGFSCKKFGDSNYSLSVKPSYTVQDRIRTVFNKSVASNLI 226

 Score = 215 (100.6 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 39/85 (45%), Positives = 58/85 (68%)

Query: 259 LLFINHRLVESTSLRKAIETVYAAYLPKNTHPFLYLSLEISPQNVDVNVHPTKHEVHFLH 318
           + FIN+RLV    LR+A+ +VY+ YLPK   PF+YL + I P  VDVNVHPTK EV FL
Sbjct: 259 IFFINNRLVTCDLLRRALNSVYSNYLPKGNRPFIYLGIVIDPAAVDVNVHPTKREVRFLS 318

Query: 319 EESILERVQQHIESKLLGSNSSRMY 343
           ++ I+E++   + ++L   ++SR +
Sbjct: 319 QDEIIEKIANQLHAELSAIDTSRTF 343
```

[2] http://blast.wustl.edu/blast/example2-14.html

```
 Score = 136 (64.7 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 40/121 (33%), Positives = 58/121 (47%)

Query: 636 LIGLPLLIDNYVPPLEGLPIFILRLATEVNWDEEKECFESLSKECAMFYSIRKQYISEES 695
           L  LPLL+  Y+P L  LP FI RL  EV+W++E+EC + + +E A+ Y        + S
Sbjct: 649 LKSLPLLLKGYIPSLVKLPFFIYRLGKEVDWEDEQECLDGILREIALLYIPDMVPKVDTS 708

Query: 696 TLSGQQSEVPGSIPNSWKWTVEHIVYKALRSHILPPKHFTEDGNILQLANLPDLYKVFERC 756
             S  + E    I      +                       +++++ANLPDLYKVFERC
Sbjct: 709 DASLSEDEKAQFINRKEHISSLLEHVLFPCIKRRFLAPRHILKDVVEIANLPDLYKVFERC 769

 Score = 93 (45.2 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 21/52 (40%), Positives = 29/52 (55%)

Query: 539 ALAQHQTKLYLLNTTKLSEELFYQILIYDFANFGVLRLSEPAPLFDLAMLAL 590
           A  QH  KL+L++   +  ELFYQI + DFANFG + L       D+ +  L
Sbjct: 549 AAIQHDLKLFLIDYGSVCYELFYQIGLTDFANFGKINLQSTNVSDDIVLYNL 600

 Score = 76 (37.4 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 17/49 (34%), Positives = 30/49 (61%)

Query: 501 INLTSVLSLQEEINEQGHEVLREMLHNHSFVGCVNPQWALAQHQTKLYL 549
           +NLTS+  L+E++++  H  L ++  N ++VG V+ +  LA  Q  L L
Sbjct: 509 VNLTSIKKLREKVDDSIHRELTDIFANLNYVGVVDEERRLAAIQHDLKL 557

 Score = 42 (22.0 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 8/26 (30%), Positives = 16/26 (61%)

Query: 609 EYIVEFLKKKAEMLADYFSLEIDEEG 634
           E I+  +   + ML +Y+S+E+  +G
Sbjct: 614 EKIISKIWDMSSMLNEYYSIELVNDG 639

 Score = 41 (21.5 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 9/33 (27%), Positives = 20/33 (60%)

Query: 365 SLTSSSTSGSSDKVYAHQMVRTDSREQKLDAFL 397
           S T++++     K   +++VR D+ + K+ +FL
Sbjct: 381 SYTTANSQLRKAKRQENKLVRIDASQAKITSFL 413

 Score = 39 (20.6 bits), Expect = 1.5e-21, Sum P(5) = 1.5e-21
 Identities = 9/27 (33%), Positives = 14/27 (51%)

Query: 411 IVTEDKTDISSGRARQQDEEMLELPAP 437
           + T+ K D  + R    + +MLE P P
Sbjct: 112 VTTKVKEDRCAWRVSYAEGKMLESPKP 138

 Score = 37 (19.7 bits), Expect = 1.7e-132, Sum P(7) = 1.7e-132
 Identities = 7/22 (31%), Positives = 13/22 (59%)

Query: 503 LTSVLSLQEEINEQGHEVLREM 524
           +TS LS  ++ N +G    R++
Sbjct: 409 ITSFLSSSQQFNFEGSSTKRQL 430
```

```
 Score = 36 (19.3 bits), Expect = 4.2e-46, Sum P(7) = 4.2e-46
 Identities = 9/40 (22%), Positives = 20/40 (50%)

Query:  14 TVVNRIAAGEVIQRPANAIKEMIENCLDAKSTSIQVIVKE 53
           TV N+  A  +I    + ++++    +D K  ++  I K+
Sbjct: 215 TVFNKSVASNLITFHISKVEDLNLESVDGKVCNLNFISKK 254

 Score = 34 (18.4 bits), Expect = 1.7e-138, Sum P(8) = 1.7e-138
 Identities = 7/20 (35%), Positives = 12/20 (60%)

Query: 242 MNGYISNANYSVKKCIFLLF 261
           ++G + N N+  KK I  +F
Sbjct: 241 VDGKVCNLNFISKKSISPIF 260

 Score = 34 (18.4 bits), Expect = 9.1e-106, Sum P(5) = 9.1e-106
 Identities = 6/23 (26%), Positives = 14/23 (60%)

Query: 209 NASTVDNIRSIFGNAVSRELIEI 231
           N +++  +R    +++ REL +I
Sbjct: 510 NLTSIKKLREKVDDSIHRELTDI 532
```

Information about the 12 HSPs is summarized in Table 10.3. The HSPs are numbered 1 to 12 as they occur above. The P-values indicated for any HSP are calculated from some Karlin–Altschul sum statistic associated with the HSP. Thus these P-values do not apply to the HSP itself, but rather to the HSP in conjunction with other HSPs with which it forms a consistent set. When more than one consistent set contains an HSP, the P-value reported for any HSP is the smallest one. Consistent sets have not been given on the standard printout, so that determining which HSPs form which consistent sets has been left to the user. An option, however, has recently been implemented in the Washington University version of BLAST 2.0 that will allow the output of consistent sets.

HSP	1	2	3	4	5	6
N	8	8	8	8	8	8
P-value	1.7e-138	1.7e-138	1.7e-138	1.7e-138	1.7e-138	1.7e-138
query span	8-229	259-343	636-756	539-590	501-549	609-634
target span	5-226	259-343	649-769	549-600	509-557	614-639

HSP	7	8	9	10	11	12
N	8	5	7	7	8	5
P-value	1.7e-138	1.5e-21	1.7e-132	4.2e-46	1.7e-138	9.1e-106
query span	365-397	411-437	503-524	14-53	242-261	209-231
target span	381-413	112-138	409-430	215-254	241-260	510-532

Table 10.3.

The eight HSPs forming the most significant set are 1, 11, 2, 7, 5, 4, 6, 3, listed in their consistent order. Notice that HSP 5 overlaps HSP 4, as shown by the HSP spans. This overlap is allowed under the default option, since there is no overlap after removing the right 12.5% of residues from HSP 5 and the left 12.5% of residues from HSP 4. The first seven HSPs are consistent; the eighth is not consistent with the previous seven, and it forms the consistent set containing HSPs 8, 5, 4, 6, 3, listed in their consistent order. HSP 9 is part of the set 1, 11, 2, 9, 4, 6, 3. HSP 10 is part of the set 10, 2, 7, 5, 4, 6, 3, and HSP 12 is part of the set 1, 12, 4, 6, 3. It might not always be so easy to find consistent sets, especially when there are hundreds of HSPs and very long HSPs. Furthermore, there may be ambiguities in that a given HSP may report an $N(= r)$ of 5, yet be consistent with two different sets of 4 HSPs. In this case BLAST reports the set with lower P-value. However, it might not be clear from the printout which set this is, and it might be necessary to calculate the significance values to find it.

10.6 Minimum Significance Lengths

10.6.1 A Correct Choice of n

When sequences are distantly related, the similarities between them might be subtle. Thus we shall not be able to detect significant similarity unless a long alignment is available. On the other hand, if sequences are very similar, then a relatively short alignment is sufficient to detect significant similarity. In this section we discuss how this issue can be put on a more rigorous foundation.

If the similarity is subtle, each aligned pair will tell us less, in terms of information, than each aligned pair in more similar sequences. This will lead us to the concept of information content per position in an alignment. The theory to be developed relates to a fixed ungapped alignment of length N.

The PAMn substitution matrix has been discussed extensively above. In this section we take for granted the evolutionary model underlying these matrices. Our analysis follows that of Altschul (1991). In particular, we assume for convenience, with Altschul, that the amino acid frequencies in the two sequences compared are the same. However, in some other respects our analysis differs from his.

The analysis of Section 10.2.4 shows that an investigator using a PAMn substitution matrix in a BLAST procedure is in effect testing the alternative hypothesis that n is the correct value to use in the evolutionary process leading to the two protein sequences compared against the null hypothesis that the appropriate value of n is $+\infty$. In this section we assume that the alternative hypothesis is correct (that is, that the correct value of n has been chosen), and in effect explore aspects of the power of the testing procedure by finding the mean length of protein sequence needed before the alternative hypothesis is accepted. In the following section we explore the effects of an incorrect choice of n.

Suppose that, in formal statistical terms, we decide to adopt a testing procedure with Type I error α. Equation (10.29) shows that the value s of the normalized score statistic S' needed to meet this P-value requirement is approximately given by $s = -\log \alpha$. From equation (10.25) the corresponding value $y_{\max}$ of $Y_{\max}$ is

$$y_{\max} = \lambda^{-1} \log \left(\frac{NK}{\alpha} \right). \tag{10.64}$$

When the alternative hypothesis is true, the mean score for the comparison of the amino acids at any position is, from (10.7),

$$\sum_{j,k} q(j,k) S(j,k) = \lambda^{-1} \sum_{j,k} q(j,k) \log \frac{q(j,k)}{p_j p_k}. \tag{10.65}$$

Equation (7.23) shows that if the mean final position in a random walk is F and the mean step size is G, the mean number of steps needed to reach the final position is F/G. This then suggests that the mean sequence length needed in the maximally scoring local alignment in order to obtain significance with Type I error α is the ratio of the expressions in (10.64) and (10.65), namely

$$\frac{\log \left(\frac{NK}{\alpha} \right)}{\sum_{j,k} q(j,k) \log \frac{q(j,k)}{p_j p_k}}. \tag{10.66}$$

Altschul (1991) calls this the “minimum significance length.” The expression (10.66) does not change if we change the base of both logarithms. The choice of the base 2 for these logarithms has an “intuitive appeal” (Altschul (1991)), since then various components in the resulting expression can be interpreted in terms of bits of information, as discussed in Appendix B.10. We thus make this choice in the following discussion, and write the ratio (10.66) as

$$\frac{\log_2 \left(\frac{NK}{\alpha} \right)}{\sum_{j,k} q(j,k) \log_2 \left(\frac{q(j,k)}{p_j p_k} \right)}. \tag{10.67}$$

We consider first the denominator in (10.67). This can be thought of as the mean of the relative support, in terms of bits, provided by one observation for the alternative hypothesis against the null hypothesis, given that the alternative hypothesis is true.

It follows that the numerator in (10.67) can be thought of as the mean total number of bits of information needed to claim that the two sequences are similar. The value of K is known from experience to be typically about 0.1, and α is typically 0.05 or 0.01. Thus the value of the numerator is largely determined by the length N, and to a close approximation is $\log_2 N$. Given the value $N = 1{,}000$, for example, this approximate numerator expression shows that about 9.97 bits of information are needed in order to claim significant similarity between the two sequences.

Our main interest, however, is not in the numerator or the denominator of (10.67), but in the ratio of the two, that is, the minimum significant length. When n is large, $q(j,k)$ is close to $p_j p_k$; the mean information per aligned pair given in the denominator is small and the minimum significant length is large. This is as expected: If null and alternative hypotheses specify quite similar probabilities for any aligned pair, many observations will in general be needed to decide between the two hypotheses. On the other hand, if n is small, the mean relative support for the alternative hypothesis provided by each aligned pair is large, and the minimum significant length is small. The limiting ($n \to 0$) values $q(j,j) = p_j$, $q(j,k) = 0$ for $j \neq k$, together with the convention that $0 \log 0 = 0$ (see Appendix B.7), show that as $n \to 0$, the denominator in (10.67), that is, the mean support from each position in favor of the alternative hypothesis, approaches $-\sum_j p_j \log_2 p_j$. If all amino acids are equally frequent, this mean support is $\log_2 20 = 4.32$, and we can think of this as 4.32 bits of information. In practice, the actual frequencies of the observed amino acids imply that a more appropriate value is about 4.17. Thus the minimum significant length is $(\log_2 N)/4.17$. If $N = 1{,}000$, this is about 2.39.

When $N = 1{,}000$ and $n = 250$, corresponding to a PAM250 substitution matrix, the probabilities $q(j,k)$ are such that each amino acid pair provides a mean of only 0.36 bits of information, and a minimum significance length of about $\log(1000)/0.36 = 9.97/0.36 = 28$ is required on average to accept the alternative hypothesis.

10.6.2 An Incorrect Choice of n

The above calculations all assume that the correct value for n has been chosen, and thus the correct alternative hypothesis probabilities $q(j,k)$ were used. In practice it is impossible to choose a unique correct value for n when using a PAM matrix, since different species in the database will have different distances from the species corresponding to the query sequence. This matter has been addressed by Altschul (1993). To illustrate some of the points at issue we suppose that there is a unique correct value m leading to a PAMm matrix, but that some incorrect value n was chosen and a PAMn matrix used instead. What does this imply?

Suppose that with the correct choice m the probability of the ordered pair (j,k) is $r(j,k)$. The mean score is then

$$\lambda^{-1} \sum_{j,k} r(j,k) \log \frac{q(j,k)}{p_j p_k}. \tag{10.68}$$

Clearly $r(j,k) = q(j,k)$ when $n = m$, and equation (1.120) then shows that the mean score is positive. More generally, the mean score is positive if n and m are close. However, as $m \to +\infty$, $r(j,k) \to p_j p_k$, and for this value of $r(j,k)$ the mean score is negative. Thus for any choice of n there will be values of m sufficiently large compared to n so that the mean score is negative. This matter is discussed further below.

In cases where the mean score (10.68) is positive, the minimal significance length is

$$\frac{\log\left(\frac{NK}{\alpha}\right)}{\sum_{j,k} r(j,k) \log \frac{q(j,k)}{p_j p_k}}. \tag{10.69}$$

This minimal length depends on $q(j,k)$, that is, on the choice of n. This choice of n may well involve substantial extrinsic guesswork, and it is thus important to assess the implications of an incorrect choice. Altschul (1991) gives examples of the effect on the minimal significance length of using scores derived from one PAM matrix when another is appropriate.

The fact that the mean (10.68) can be negative requires some discussion. Negative means arise when m is sufficiently large compared to n, that is, when the two species being compared diverged a long time in the past relative to the time assumed by the PAM matrix used in the analysis. In this case the data are better explained by assuming no similarity between the two sequences than by assuming a close similarity between the two sequences. The more negative this mean, the more likely it is that the null hypothesis will be accepted, and in the limit $m \to +\infty$, when $r(j,k) = p_j p_k$, the probability of rejecting the null hypothesis is equal to the chosen Type I error.

As an example of this effect, if in the simple symmetric model of Section 6.5.4 the value $n = 100$ is chosen, the mean score (10.68) is negative when m is 193 or more.

These observations indicate the perils of deciding on too small a value of n. Whereas a correctly chosen small value of n leads to shorter minimal significance lengths, as discussed above, an incorrectly small choice may lead to the possibility that a real similarity between the two sequences will not be picked up. The practice sometimes adopted of using a variety of substitution matrices to overcome this problem must be viewed with some caution, particularly in the light of the multiple testing problem discussed in Section 3.11.

10.7 BLAST: A Parametric or a Non-parametric Test?

In parametric tests the test statistic is found from likelihood ratio arguments, as discussed in Chapter 9. By contrast, the test statistic in a non-parametric test is often found on reasonable but nevertheless arbitrary grounds, as was, for example, the non-parametric Mann–Whitney test statistic discussed in Section 3.8.2.

Many of the calculations and arguments used in the immediately preceding sections derive from the derivation of the score $S(j, k)$ in a substitution matrix from likelihood ratio arguments: See, for example, equations (10.6) and (10.7). In this sense the BLAST testing theory can be thought of as a parametric procedure deriving from the likelihood ratio theory in Section 9.2.1.

The assumptions made in this theory are, however, subject to debate. For example, Benner et al. (1994) claim that the time homogeneity assumption implicit in the calculations cannot be sustained, claiming, for example, that the genetic code influenced substitutions earlier in time and various chemical properties influenced substitutions more recently. Thus comparisons of distantly related species can be problematic. Even in the comparison of more closely related species, it is not clear that a uniform set of rules governs substitutions. Further, if the data in a large database come from a collection of species whose respective evolutionary divergence times might differ widely, the concept of a uniformly correct choice of n (see Section 10.6) is not meaningful.

Even if these, and similar claims are true, the statistical aspects of the BLAST procedure are still valid, in the sense that the P-value calculations are still correct. The P-value calculations take the scores in the substitution matrix as given, so that even if these scores were chosen in any more or less reasonable way, rather than from theoretical deductions using some evolutionary Markov chain and likelihood ratio theory, no problems arise with the correctness of the P-value calculations. In this sense the BLAST testing process can be thought of as a non-parametric procedure, where the choice of test statistic does not derive from a likelihood ratio or any other optimality argument but is chosen instead on commonsense grounds. On the other hand, if the various assumptions implicit in finding a substitution matrix from likelihood ratio arguments are not correct, some of the theory in the preceding sections, particularly that associated with the optimal choice of n for a PAMn matrix, needs amendment.

10.8 Gapped BLAST and PSI BLAST

10.8.1 Gapped BLAST

In this section we outline two important generalizations that have been made and are incorporated in current BLAST implementations.

The first generalization allows gaps in the sequence alignments (Altschul et al. (1997)). To outline this generalization we first recall a result from the ungapped theory, namely that in the case of two unaligned sequences of respective lengths N_1 and N_2, the approximate mean and variance of the test statistic $Y_{\max}$, given in (10.30) and (10.31), are

$$\lambda^{-1}(\log(KN_1N_2)) \quad \text{and} \quad \pi^2/(6\lambda^2) \tag{10.70}$$

respectively, and that $Y_{\max}$ has an (approximate) distribution given implicitly by (10.43).

Suppose now that gaps are allowed in the alignment of the two sequences, with some chosen linear gap penalty. In the comparison of the two sequences there will be some maximum score $Y_{\max}^{(\text{gapped})}$, the maximum score over all possible gapped alignments. The null hypothesis probability distribution of $Y_{\max}^{(\text{gapped})}$ is determined by the substitution matrix used and the gap penalty chosen. However, this null hypothesis distribution is not easy to find, and Altschul et al. (1997) follow an empirical approach to estimating it, using simulation results of Altschul and Gish (1996).

These simulations were carried out using various substitution matrices and various gap penalties. In the case of the BLOSUM62 substitution matrix, the gap penalty used was chosen to be $12 + k$ for a gap of size k. Two independent amino acid sequences, of respective lengths N_1 and N_2, were generated at random, using the amino acid probabilities given by Robinson and Robinson (1991). From these sequences the highest score, denoted here y_1, the observed value of $Y_{\max}^{(\text{gapped})}$, was found. This procedure was then repeated a large number n of times (n =10,000 in their simulations), yielding n observed highest scores $y_1, y_2, \ldots, y_n$. The mean and variance of $Y_{\max}^{(\text{gapped})}$ were then estimated using $\bar{y}$ and s^2 (defined in (3.6) with a change of notation from x_i to y_i) respectively.

The approximation is then made that the distribution of $Y_{\max}^{(\text{gapped})}$ is of the same the form (10.43) as that arising in the ungapped case, with revised values for K and λ.

The method of moments procedure, discussed in Section 8.4, is used to estimate the revised values of K and λ, using the method of moments equations

$$\bar{y} = \hat{\lambda}^{-1}(\log(\hat{K}N_1N_2), s^2 = \quad \pi^2/(6\hat{\lambda}^2), \tag{10.71}$$

derived from (8.35) and (10.70). The solution of these equations is

$$\hat{K} = (N_1N_2)^{-1}e^{\bar{y}\hat{\lambda}}, \quad \hat{\lambda} = \pi/(s\sqrt{6}). \tag{10.72}$$

This procedure was then repeated for a number of (N_1, N_2) combinations. There is no guarantee that the estimates of K and λ are independent of N_1 and N_2. The value of $\hat{\lambda}$ does however appear to be approximately independent of N_1 and N_2, being about 85% of the corresponding value in the ungapped case. The values found for $\hat{K}$ do depend on N_1 and N_2, but carrying out edge corrections as in Section 10.3.3 does appear to overcome this problem to a large extent.

There is also no guarantee that the complete distribution of $Y_{\max}^{(\text{gapped})}$ is close to that of the ungapped statistic $Y_{\max}$, even with a change of the parameters K and λ as just described. To a first approximation, however, this appears to be the case for the BLOSUM62 matrix. With this degree of empirical support, the gapped case is handled as in the ungapped case with revised values of K and λ.

Simulations with a PAM250 matrix (with a gap penalty of $15 + 3k$) lead to similar conclusions, as do simulations using the sum statistics described in Section 10.2.8. Since the BLOSUM62 and the PAM250 substitution matrices are used often, these conclusions are useful in practice. Since only very small P-values are usually of interest in a BLAST search, comparatively small inaccuracies in the above approximations are probably not important.

Gapped BLAST calculations at NCBI no longer use the Karlin-Altschul sum statistic, so the corresponding printouts do not show the N column in the BLAST printout.

The approach described above and currently implemented depends on simulation results, carried out necessarily for a restricted range of cases. However, generalizations of the theory to cover the case of gaps have recently been made: see Mott and Tribe (1999), Siegmund and Yakir (2000), Storey and Siegmund (2001), and Chan (2003). Storey and Siegmund show that if a penalty of δ is assigned to each gap in the alignment of two sequences, then (10.45) should be replaced by

$$E' = N_1 N_2 K e^{-\lambda y_{\max}} \Big(1 - \frac{T}{e^{\theta^* \delta} - 1} \Big), \tag{10.73}$$

for a constant T whose explicit form we do not give here. The choice $\delta = +\infty$ in effect allows no gaps, and in this case (10.73) reduces to (10.45). Chan (2003) considers the case of an arbitrary non-decreasing gap penalty, and using a generalization of the mgf equation (10.3) that incorporates this penalty, finds a sharp upper bound for the P-value associated with any observed value of $y_{\max}$.

10.8.2 PSI BLAST

A second generalization is PSI (position specific iterated) BLAST. In regular BLAST a fixed substitution matrix is used to score positions in alignments, regardless of the position in the query sequence. Substitution matrices are trained on data mainly from the alignment of well conserved functional domains in protein coding genes, and the procedure relies on one matrix to provide, on average, the most meaningful scores for all positions in the query sequence simultaneously. In PSI-BLAST, the procedure using a standard substitution matrix is used as a first step. The sequences that are found are then used to derive a separate scoring scheme for each position in the query sequence. This new scoring scheme is then used to perform a second BLAST search, which can be more sensitive and thus find subtler homology than does the first. The sequences returned on the second iteration can then be used to derive a scoring scheme again, and perform a third round, which can be more sensitive than the second. This procedure can be iterated until no further iteration seems useful.

An entire substitution matrix is not derived for each position in the query sequence. Since the base in the query sequence does not change, what will be derived for each position is essentially the one row in the matrix corresponding to the particular base at that position in the query sequence. If the same base exists in two or more positions in the sequence, each position will still get its own (most likely) unique scoring scheme. This leads to the term "position specific iterated" (PSI) BLAST.

An outline of the original procedure, which is carried out in association with gapped BLAST, is as follows. The query sequence is first compared to the data base using standard BLAST methods, and all database sequence segments having a sufficiently close similarity with the query (for example having a value of "Expect" less than 0.01) are noted. Various data-trimming procedures are now carried out; for example, only one copy of closely similar database segments are retained (found by using arguments similar to those leading to the Henikoff and Henikoff procedure of Section 6.5.2).

Consider now some site in the query sequence. This site will be aligned with some collection of the remaining database segments, and in general some interval of query sequence sites around this site will also align to these segments. From this collection of sites a frequency f_i of amino acid i is calculated. These frequencies are to be used as a basis for estimating the frequency Q_i of amino acid i at this site.

The original PSI-BLAST implementation as described in Altschul et al. (1997) estimated Q_i by using pseudocounts. The pseudocount frequencies g_i are defined by

$$g_i = \sum_j f_j q(i,j)/p_j, \tag{10.74}$$

where $q(i,j)$ is a frequency generically of the "target" form (10.8). Q_i is then defined as a linear combination

$$Q_i = \frac{\alpha f_i + \beta g_i}{\alpha + \beta}. \tag{10.75}$$

While in standard BLAST $q(i,j) = q(j,i)$, this equality no longer occurs automatically in the iterations of PSI-BLAST. This implies that the equation $\sum_i g_i = 1$ no longer necessarily holds, so that the g_i do not necessarily form a probability distribution. Because of this problem a new form of PSI-BLAST, described in Schäffer et al. (2001) has been implemented. In this implementation, Q_i is in effect defined as

$$Q_i = \frac{\alpha f_i + \beta \sum_j f_j p(i,j)/p_j}{\alpha + \beta}, \tag{10.76}$$

where p_i is the background frequency of amino acid i and $p(i,j)$ is the frequency with which amino acids i and j are aligned through evolutionary descent. With this definition the required equality $\sum_i Q_i = 1$ does hold.

The logarithm of the ratio Q_k/P_k is now used in a manner similar to that on the right-hand side in (10.10) to form a score to be used in the iterated PSI-BLAST process.

Another generalization, not currently implemented in BLAST, is to the case of Markov-dependent sequences, the theory for which is developed by Karlin and Dembo (1992). However, the theory for this generalization, and the full theory for the other generalizations referred to above, is beyond that appropriate for an introductory book.

10.9 Relation to Sequential Analysis

There are many similarities between the BLAST calculations given in this chapter and sequential analysis calculations discussed in Section 9.9. First and foremost, the central BLAST parameter λ ($= \theta^*$) was first introduced into probability theory in the context of sequential analysis, being used in that theory to calculate power curves (see equations (9.62) and (9.64)), as well as mean sample size (see equation (9.69)). Second, both sequential analysis and BLAST theory center on running sums of iid random variables, and further, the random variables in both cases are either logarithms of likelihood ratios or multiples of logarithms of likelihood ratios.

It is therefore interesting to compare further the calculations deriving from (10.66) with the analogous calculation for a sequential test of hypothesis. If the alternative hypothesis is true, the mean step size in the sequential procedure defined by (9.55) is

$$\sum_y p(y;\xi_1)\log\left(\frac{p(y;\xi_1)}{p(y;\xi_0)}\right).$$

From (9.55), the accumulated sum in the sequential procedure necessary to reject the null hypothesis is $\log((1-\beta)/\alpha)$, where α and β are the Type I and Type II errors, respectively. If these errors are both small, as is normally the case, this is close to $\log(1/\alpha)$. If we argue as in the derivation of the ratio (10.66) above, the mean number of observations needed to reject the null hypothesis when the alternative hypothesis is true, in a test with Type I error α, would be the ratio

$$\frac{\log(1/\alpha)}{\sum_y p(y;\xi_1)\log\left(\frac{p(y;\xi_1)}{p(y;\xi_0)}\right)}. \tag{10.77}$$

If we identify the observation y in a sequential test with the pair (j,k) in a sequence comparison, the denominators in the two expressions (10.66) and (10.77) are identical. The comparison between the two expressions thus concerns only their respective numerators. The numerator in (10.66) can be written as $\log(1/\alpha) + \log(N_1 N_2 K)$. The difference between the two numerators is, then, the additive factor $\log(N_1 N_2 K)$. This factor arises because in the BLAST procedure the test statistic is essentially the maximum of $N_1 N_2/A$ iid geometric-like random variables, and the mean of such a maximum, like the mean of the maximum of n iid geometric random variables given in equation (2.118), is approximately $\log(N_1 N_2 K)$, as shown in (10.31). This comparison shows how much more stringent a test based on a maximal test statistic must be compared to one based, in the sequential procedure, on the typical value of a statistic. Once allowance for this difference is made, the similarity between the two procedures becomes apparent.

A second connection between sequential analysis and BLAST testing derives from the comparison of the denominator in the sequential analysis expression (9.70) and the denominator in the BLAST expression (10.69). In the sequential analysis case the form of the correct probability distribution $Q(y)$ of Y differs from that assumed under the null and alternative hypotheses. In the BLAST case the parallel comment might be, for example, that the elements in the substitution matrix were calculated from the evolutionary process leading to some PAM matrix, whereas some quite different evolutionary model might be appropriate.

A further connection between the BLAST and the sequential analysis testing procedures is that in both cases the step size in the testing procedure depends implicitly on some alternative hypothesis. In this respect both procedures differ from the (fixed-sample-size) test of Section 3.4.1 for the parameter p in a binomial distribution, where the testing procedure is independent of the alternative hypothesis value of p (so long as it exceeds the null hypothesis value).

Despite these connections between BLAST and the sequential testing procedure, the two procedures are rather different, and in some respects the BLAST procedure is more like the fixed sample size test. For example, the sample size is in effect fixed in advance and the test does not rely on achieving some specified Type II error.

Problems

10.1. Consider the calculation that led to equation (10.5). Use the path decomposition method to do the analogous calculation for the probability u that the generalized random walk under consideration reaches -1 as its first ladder point. Check that $u + v = 1$.

10.2. For the simple random walk of Section 7.2 the value of θ^* is given in (7.7), the value of C is $1 - e^{-\theta^*}$, and the value of A is $(q - p)^{-1}$. From this, the value of K, calculated from equation (10.18), is $(q-p)(e^{-\theta^*} - e^{-2\theta^*})$. Making the change of notation $\theta^* = \lambda$, check that both equations (10.23) and (10.24) give this value.

10.3. (This and the following problems refer to the symmetric PAM matrix discussed in Section 6.5.4.) the case $C = 1$ corresponding to the value $n = 259$ leads to a mean step size, when the alternative hypothesis is true, of 0.446 (see equation (6.36)). BLAST theory shows that this value should also be given by the expression $\lambda^{-1}H$ (see (10.13)). Use the values for $q(j,k)$ and $q(j,j)$ given in (6.35), the values $p_j = p'_k = 0.05$ in the expression (10.11) to compute H, and equation (10.3) to compute λ (in the case $S(j,j) = 12$, $S(j,k) = -1$ $(j \neq k)$), to verify this.

10.4. *Continuation.* Show mathematically that the alternative hypothesis mean size in any simple symmetric PAM model (that is, for any value of n), is equal to the value of $\lambda^{-1}H$ for that model.

10.5. *Continuation.* Use the expression (6.32) for each diagonal element in the substitution matrix for the simple symmetric PAM model and the value -1 for each off-diagonal element, together with equation (10.3), to show that in the simple symmetric PAM model the value of λ is $-\log\bigl(1-\bigl(\frac{94}{95}\bigr)^n\bigr)$. If the value of λ is 0.320 (as in the printout of the example of Section 10.5.1, what is the corresponding value of n?

10.6. *Continuation.* Suppose that the PAM model of the "simple symmetric" example of Section 6.5.4, for which in particular $p_j = p'_k = 0.05$, leads to a substitution matrix in which $S(j,j) = 10$ $(j = 1, 2, \ldots, 20)$ and $S(j,k) = -1$ $(j \neq k)$.

(i) Use equation (10.3) to find the associated value of λ. (This will require numerical methods.)

(ii) From the result of (i), use equation (10.24) to find K.

(iii) Use equation (10.8) to find the (common) values of $q(j,j)$ $(j = 1, 2, \ldots, 20)$ and the (common) values of $q(j,k)$ $(j \neq k)$.

(iv) From the results of (ii) and (iii), find the relative entropy H defined in (10.11).

(v) Use equations (6.28) and (6.29) to find the value of n implied by the values of $q(j,j)$ and $q(j,k)$ found in (iii) above.

(vi) Use the value of n found in (v) in the expression (6.31) to confirm the ratio -10 for $S(j,j)/S(j,k)$.

10.7. *Continuation.* Repeat Problem 10.6 with the value 10 for $S(j,j)$ replaced by (i) 6, 8, 12, and 14, with $S(j,k) = -1$ $(j \neq k)$ and $p_j = p'_j = 0.05$

(as in Problem 10.6). Compare your values of λ, K, and H with those on BLAST printouts.

10.8. *Continuation.* Repeat Problem 10.6 with $S(j,j)$ replaced by 20 and $S(j,k)$ replaced by -2. Comment on the similarities and differences between your calculations and those of Problem 10.6.

10.9. *Continuation.* Suppose that the diagonal elements in a simple symmetric PAM matrix all take the value S and all off-diagonal elements take the value -1. If $\lambda = 0.320$ (as in the printout of Section 10.5.1), find the value of S. From this, use equation (6.32) to find the value of n (in the PAMn matrix). Also, use equation (10.24) to find the value of K, and compare this with the value in the printout of Section 10.5.1.

10.10. *Continuation.* Suppose that in the simple symmetric example of Section 6.5.4 the value $n = 50$ is chosen to calculate the simple PAM substitution matrix. Find the values of the true value m for this model for which the mean score (10.68) is negative.

10.11. Suppose that only two amino acids "X" and "Y" exist, occurring with respective frequencies 0.6 and 0.4. Suppose that a PAM matrix is used in a sequence alignment and that the match probabilities corresponding to this matrix are $q(X,X) = 0.46$, $q(X,Y) = 0.28$, $q(Y,Y) = 0.26$. Compute the mean score (10.68) in the cases (i) $r(X,X) = 0.38$, $r(X,Y) = 0.44$, $r(Y,Y) = 0.18$, (ii) $r(X,X) = 0.40$, $r(X,Y) = 0.40$, $r(Y,Y) = 0.20$, (iii) $r(X,X) = 0.42$, $r(X,Y) = 0.36$, $r(Y,Y) = 0.22$, (iv) $r(X,X) = 0.44$, $r(X,Y) = 0.32$, $r(Y,Y) = 0.24$. Comment on your answers.

10.12. Use the BLOSUM62 substitution matrix of Table 6.7 to check the score 70 given for the Maize Glutathione match in the BLAST printout of Section 10.5.

10.13. This problem refers to Table 1 of Karlin and Altschul (1993). Given the values of λ, K, and $N(= N_1N_2)$ referred to in their paper, confirm that the three normalized scores listed can be derived from the three corresponding scores listed, using equation (10.25).

10.14. This problem refers to Table 2 of Karlin and Altschul (1993). Given the values of "Score," λ, K, and $N(= N_1N_2)$ referred to in their paper, confirm their calculations for the various normalized scores and the various segment and sum P-values.