BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 22 2005, pages 4125–4132
doi:10.1093/bioinformatics/bti658

Sequence analysis

Optimal word sizes for dissimilarity measures and estimation of the degree of dissimilarity between DNA sequences

Tiee-Jian Wu[1,*], Ying-Hsueh Huang[2,3] and Lung-An Li[2]

[1]Department of Statistics, National Cheng-Kung University, Tainan, Taiwan 70101, [2]Institute of Statistical Science, Academia Sinica, Taipei, Taiwan 11529 and [3]Institute of Bioinformatics, National Yang-Ming University, Taipei, Taiwan 11221

Received on May 25, 2005; revised on August 24, 2005; accepted on August 31, 2005
Advance Access publication September 6, 2005

ABSTRACT

Motivation: Several measures of DNA sequence dissimilarity have been developed. The purpose of this paper is 3-fold. Firstly, we compare the performance of several word-based or alignment-based methods. Secondly, we give a general guideline for choosing the window size and determining the optimal word sizes for several word-based measures at different window sizes. Thirdly, we use a large-scale simulation method to simulate data from the distribution of SK–LD (symmetric Kullback–Leibler discrepancy). These simulated data can be used to estimate the degree of dissimilarity β between any pair of DNA sequences.

Results: Our study shows (1) for whole sequence similiarity/dissimilarity identification the window size taken should be as large as possible, but probably not >3000, as restricted by CPU time in practice, (2) for each measure the optimal word size increases with window size, (3) when the optimal word size is used, SK–LD performance is superior in both simulation and real data analysis, (4) the estimate $\hat{\beta}$ of β based on SK–LD can be used to filter out quickly a large number of dissimilar sequences and speed alignment-based database search for similar sequences and (5) $\hat{\beta}$ is also applicable in local similarity comparison situations. For example, it can help in selecting oligo probes with high specificity and, therefore, has potential in probe design for microarrays.

Availability: The algorithm SK–LD, estimate $\hat{\beta}$ and simulation software are implemented in MATLAB code, and are available at http://www.stat.ncku.edu.tw/tjwu

Contact: tjwu@stat.ncku.edu.tw

Supplementary information: Tables A1–A3, and Remarks 1–11 at http://www.stat.ncku.edu.tw/tjwu

1 INTRODUCTION

In the past two decades, the number of DNA sequence records has grown exponentially over time. The characterization of new sequence data presents the biologist with many methods of sequence comparison. Several measures of DNA sequence similarity/dissimilarity have been developed in the past. The purpose of this paper is 3-fold. First, we compare the performance of several word-based or alignment-based methods. Second, we give a general guideline for choosing the (sliding) window size and determine the optimal word sizes for several word-based measures at different window sizes. Third, we approximate the distribution of the SK–LD (symmetric Kullback–Leibler discrepancy) I_n, where n denotes the word size, and use such approximation to estimate the degree of dissimilarity, denoted by β herein, between any pair of DNA sequences.

Throughout we focus on the U-I (uniform-independent) model of base composition, where the probability of encountering any one of the four bases (or letters), A, C, G and T, is taken to be 0.25 independent of the other bases. This model is appropriate because all sequences we generate in this paper can be treated as sequences drawn from the unlimited nucleotide virtual pool, see Sege and Saxberg (1982) for details of this viewpoint. Furthermore, the independence of bases is an approximation to the actual dependence in DNA sequences. Arratia *et al*. (1990) evaluated this approximation and found it to be quite good. Section 2 briefly reviews several word-based dissimilarity measures and shows that the time complexity of computing SK–LD is significantly lower than that of an alignment-based method. Section 3 contains the main results. All results are obtained through extensive simulations that involve generating a large number of pairs of m–s (mother–son) sequences, where a mother sequence is randomly generated according to the U-I model of base composition, and the son sequence is a mutated version of the mother sequence. The type of mutation considered is the point mutation including three equally likely operations, namely, inserting, deleting and substituting a base at a randomly selected position of the mother sequence, while the selection of a base for insertion and substitution is done uniformly over the possible bases. The point mutation is the most common type of copying error and are actual chemical changes to the structure of the constituent DNA. See, e.g. Alberts *et al*. (1994) for details. In Section 3.1, we use the Spearman's rank statistic to determine the optimal word size for the word-based methods ED (Euclidean distance) d_n^2, SED (standardized Euclidean distance) S_n^2, I_n and SC–RD [symmetric Cressie–Read family of discrepancies (Cressie and Read, 1984)] I_n^λ, respectively, at window size varying from 10 to 6100. We then compare their performance with Hamming distance (see, e.g. Pinheiro *et al*., 2000) and the benchmark method BLAST [Basic Local Alignment Search Tool (Altschul *et al*., 1990, 1997)] that requires sequence alignment. The results are shown in Figure 2. Figures 2a and b show, when the optimal word size is used, SK–LD performs the best among all the aforementioned methods. Figure 2c shows, for whole sequence similarity/dissimilaity comparison, the window size taken should be as large as possible (but probably not >3000, as restricted by CPU time in practice). A practical

[*]To whom correspondence should be addressed.

implementation is to let the window size be the minimum of 3000 and the lengths of the two DNA sequences under comparison. Section 3.2 constructs Table A1 at http://www.stat.ncku.edu.tw/tjwu that approximates the percentile point of the distribution of SK–LD (at optimal word size) based on a large set of simulated data. This table can be used to obtain estimate $\hat{\beta}$ of β between any pair of DNA sequences. Real data analysis in Section 4 shows $\hat{\beta}$ performance that is superior with a wide range of real sequences having U-I, skewed or first order to fifth order Markov chain base compositions, and is very robust to the misspecification of the model of base composition in practically realistic settings. In particular, $\hat{\beta}$ performs more favorably than BLAST and the alignment-free PSM (probabilistic similar measure) of Pham and Zuegg (2004) and improves the combined KL-D in Wu *et al*. (2001). Moreover, a complete genome analysis in Section 4.4 shows $\hat{\beta}$ is also applicable in local similarity comparison situations. Specifically, it shows $\hat{\beta}$ can help in selecting oligo probes with high specificity and has potential in probe design for gene expression microarrays. Finally, some remarks (Remarks 1–11) are given at http://www.stat.ncku.edu.tw/tjwu

2 DISSIMILARITY MEASURES

Many rigorous DNA sequence comparison algorithms like FASTA (Pearson and Lipman, 1988; Pearson, 1990) and BLAST involve sequence alignment at some stage and become computationally prohibitive when comparison against a large database is required. Comparison algorithms using word frequencies as a measure of dissimilarity do not require sequence alignment. They measure the frequencies of words within a DNA sequence and then compare these frequencies between DNA sequences using statistical distances. In this way they can determine the relative dissimilarity in a large database of DNA sequences very rapidly. They have already been used as pre-selection filters to filter out highly dissimilar sequences and speed alignment-based database search for similar sequences. These filtration methods are currently being increasingly explored to optimize database search and gradually being incorporated in widely used bioinformatics applications. It is noteworthy that these word-based algorithms can also find some new functional similarities or dissimilarities that are invisible to other algorithms like FASTA (Blaisdell, 1989a; Hide *et al*., 1994) and are useful in detection of coding regions (Fichant and Gautier, 1987) and evolutionary tree reconstruction (Blaisdell, 1989a,b). Several word-based algorithms (Blaisdell, 1986, 1989a; Cressie and Read, 1984; Hide *et al*., 1994; Pevzner, 1992a,b; Torney *et al*., 1990; Wu *et al*., 1997, 2001, among others) have been developed. Vinga and Almeida (2003) review these algorithms and predict that the next few years will see some of them become widely used for functional annotation and phylogenetic study.

An n-word is a subsequence of n adjacent letters. Let α be any n-word within a strand of DNA of length $m+n-1$. Define $X_i=1$ if α begins at position i and $X_i=0$ otherwise. Also, define $N_\alpha=\sum_{i=1}^m X_i$ as the frequency of α. Given two strands of DNA sequences Q and L (for the query and a library sequence in a database), let $V_{L,n}=(N_{L,\alpha_1}/m,\ldots,N_{L,\alpha_{4^n}}/m)$ be the vector of relative frequencies of n-words over a segment W_L, which is a window of length $m+n-1$ from the sequence L, where $\alpha_1,\ldots,\alpha_{4^n}$ are all of the possible n-words. Let $V_{Q,n}=(N_{Q,\alpha_1}/m,\ldots,N_{Q,\alpha_{4^n}}/m)$ and W_Q be defined similarly for Q. Thus, $Z'_n=(z_{n1},\ldots,z_{n4^n})=V_{L,n}-V_{Q,n}$ is an expression of the dissimilarity of W_L and W_Q with respect to word composition. In what follows, W_L and W_Q are shifted over L and Q, respectively. A distance (say, window distance) is taken for each pair $W=(W_L,W_Q)$. The distance between L and Q is taken to be the minimum of all window distances. The (squared) ED

$$d_n^2=\min_W d_{n,W}^2 \quad \text{with } d_{n,W}^2=Z'_n\cdot Z'_n=\sum_{i=1}^{4^n} z_{ni}^2 \tag{1}$$

is the simplest distance (cf., e.g. Pevzner, 1992a,b; Torney *et al*., 1990). It can be improved upon when some information on the variance/covariance is known. A distance better than the ED is the SED (cf., Wu *et al*., 1997)

$$S_n^2=\min_W S_{n,W}^2 \quad \text{with } S_{n,W}^2=\sum_{i=1}^{4^n} z_{ni}^2/\mathrm{Var}(m^{-1}N_{\alpha_i}) \tag{2}$$

that is, the variances of frequencies of n-words are accounted for. See Gentleman and Mullin (1989) or Wu *et al*. (1997) for the evaluation of variances. Next, if we view both $V_{L,n}$ and $V_{Q,n}$ as discrete distributions over the *4*^{n} possible n-words, then the Cressie–Read family of discrepancy between $V_{L,n}$ and $V_{Q,n}$ is defined by $I^\lambda_{L,Q}(=I^\lambda(V_{L,n},V_{Q,n}))=\{\lambda(\lambda+1)\}^{-1}\sum_{i=1}^{4^n}\{(N_{L,\alpha_i}/N_{Q,\alpha_i})^\lambda-1\}N_{L,\alpha_i}/m$, $\infty<\lambda<\infty$, where the values at $\lambda=0$, -1 are defined by continuity. To avoid the possibility that $I^\lambda_{L,Q}=\infty$, we need to modify $I^\lambda_{L,Q}$. By the approach in Frith *et al*. (2004), the discrepancy becomes

$$\tilde{I}^\lambda_{L,Q}=\{(m+0.5)\lambda(\lambda+1)\}^{-1} \times\sum_{i=1}^{4^n}\{\{(N_{L,\alpha_i}+\varepsilon_n)/(N_{Q,\alpha_i}+\varepsilon_n)\}^\lambda-1\}(N_{L,\alpha_i}+\varepsilon_n), \tag{3}$$

where $\varepsilon_n=0.5\times4^{-n}$. Since $\tilde{I}^\lambda_{L,Q}$ is not symmetric in its arguments, it is better to use the SC–RD defined by

$$I_n^\lambda=\min_W I_{n,W}^\lambda \quad \text{with } I_{n,W}^\lambda=(\tilde{I}^\lambda_{L,Q}+\tilde{I}^\lambda_{Q,L})/2, \tag{4}$$

which is symmetric. Notice that the information theory-based measure SK–LD I_n is a member of the SC–RD family. In fact, $I_n=I_n^0$ with

$$\tilde{I}^0_{L,Q}=(m+0.5)^{-1}\sum_{i=1}^{4^n}\{\log\{(N_{L,\alpha_i}+\varepsilon_n)/(N_{Q,\alpha_i}+\varepsilon_n)\}\}(N_{L,\alpha_i}+\varepsilon_n). \tag{5}$$

Furthermore, I_n^λ and d_n^2 can be computed fairly fast because they do not depend on the model of base composition and do not require computation of variances. Wu *et al*. (2001) showed that both $\tilde{I}^0_{L,Q}$ and S_n^2 have better sensitivity and selectivity than d_n^2. Now, let l_Q, l_L and l denote the lengths of the query, library and window, respectively. It can be shown (Remark 1) that for computing both I_n and d_n^2 the time complexity is $O(l_Q l_L l^{-1})$, which becomes the linear $O(\max\{l_Q,l_L\})$ if we take $l=\min\{l_Q,l_L\}$. This time complexity is significantly lower than that of an alignment-based method, whose time complexity of finding the smallest penalty alignment (Waterman, 1989) is the quadratic $O(l_Q l_L)$. We report the actual CPU time and memory space required on a PC in computing SK–LD over a test dataset in Section 4.2.

3 METHODOLOGY

In Section 1 we have described in detail the way of generating pairs of m–s sequences. In order to use Equations (1)–(5) to compute d_n^2, S_n^2, I_n and I_n^λ between a mother and her son, we take the window size $l = \min\{l_m, l_s\}$ where l_m and l_s denote the length of the mother and her son, respectively. The difference between l_m and l_s is small (evidently, $El = El_s = l_m$). Nevertheless, the window is shifted from left to right over the longer sequence between the mother and her son, and we take 90% overlap on the windows throughout our simulation.

3.1 Comparison of measures and their optimal word sizes

For several particular examples of real genomic databases Hide *et al*. (1994) give a heuristic solution to the problem of optimizing the word size for d_n^2. In this section, we shall give a simple and general solution to this problem for each of d_n^2, S_n^2, I_n and I_n^λ. We generate 5000 independent mother sequences of the same length, while the lengths considered are 10, 20, 50, 100 + 50j, $j = 0, 1, 2, \ldots, 120$ bases, respectively. For each mother, we generate a son sequence at each of the following 100 mutation rates: 1%, 2%, ..., 100%, where a mutation rate γ% means the son is obtained by randomly selecting γ% of the bases in the mother sequence for mutation and other bases are unchanged. For an alternative way of generating m-s pairs, see Remark 2.

It is quite worth analyzing the sensitivity of word-based dissimilarity measures to mutation, window size and word size. We have done analysis on the sample mean and variance of scores for ED [see also Torney *et al*. (1990)], SED, SK–LD and some other members of SC–RD family. Since the lessons learned are the same, we shall just present the results for SK–LD here. Figure 1a shows the sample mean of the 5000 SK-LD scores, resulting from the above 5000 independent m–s pairs, at every mutation rate $\gamma\% = 1\%, 2\%, \ldots, 100\%$ for window sizes 250 and 1600, and word size $n = 2, 4, \ldots$ and 10 (results for other cases of window size and n are similar), where for each window size and n the mean score is normalized—divided by the mean score at 100% mutation. It shows that the window size has a smaller influence than n on the SK–LD mean score. For each n as γ increases, the mean score increases. The mean score increases rapidly when γ is small and slowly when γ is large, and this phenomenon becomes more distinct as n gets larger. In terms of the slope of the curves, the larger the n the larger the slope at smaller γ, and the smaller the n the larger the slope at larger γ. Therefore, from the viewpoint of discerning the mean scores, longer word sizes are better for lower mutation rates and shorter word sizes are better for higher mutation rates. On the other hand, Figure 1b shows the logarithm of the sample SD of the 5000 SK–LD scores at every word size $n = 1,2, \ldots, 20$ for window size 600 and γ varying from 5 to 100%, where for each n the SD of scores is normalized—divided by the SD of scores at 100% mutation. It shows that for larger n the SD tends to be smaller at larger γ, and for smaller n the SD tends to be smaller at smaller γ (results for other cases of window size and γ are similar). Therefore, from the viewpoint of minimizing the SD of scores, longer word sizes are better for higher mutation rates and shorter word sizes are better for lower mutation rates. These two conflicting viewpoints demonstrate that the choice of word size implies a trade-off between discerning the mean and minimizing the variance, i.e. between reducing systematic and random errors. This implies that moderate word sizes are preferable because they balance the systematic and random errors and lead to smaller overall error over the whole spectrum of mutation rates $\gamma\% = 1\%, 2\%, \ldots, 100\%$. In what follows we shall describe a method based on rank statistics that exactly chooses moderate word sizes as optimal solutions.

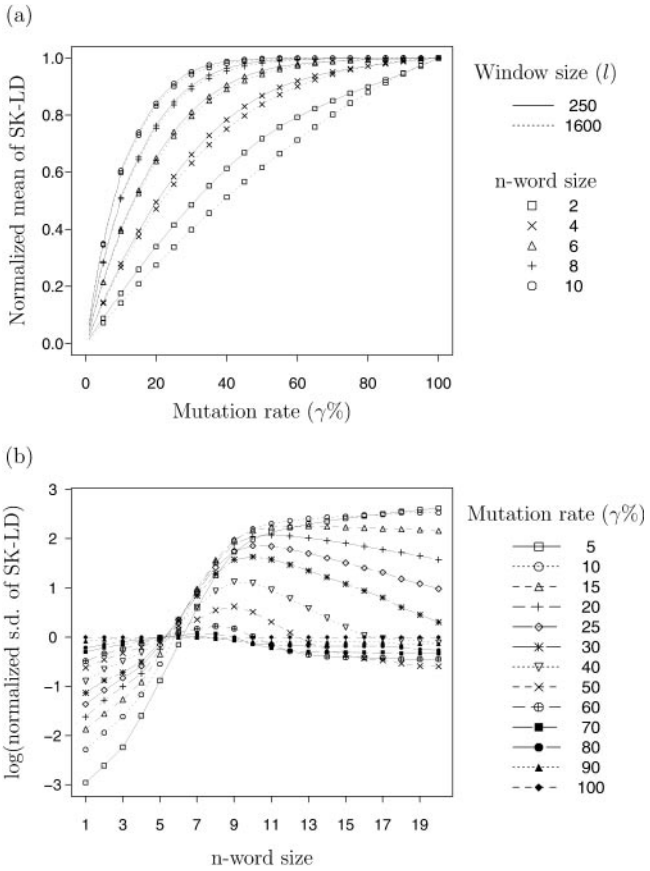

Fig. 1. (**a**) Relation between the sample mean of 5000 SK–LD scores and mutation rate $\gamma\% = 1\%, 2\%, \ldots, 100\%$. For each window size and word size n the mean scores are normalized—divided by the mean score at 100% mutation. (**b**) gives the relations between $1 \le n \le 20$ and the logarithm of sample SD of 5000 SK–LD scores at window size 600 and $\gamma = 5\%, 10\%, \ldots, 100\%$, where for each n the SD of scores is normalized—divided by the mean and SD of scores at 100% mutation. Similar patterns present for other settings of n, γ or window size in (a) and (b).

Let $X_{i\gamma}$ denote the dissimilarity score between the i-th mother and her son at mutation rate γ% and $R_{i\gamma}$ the rank of $X_{i\gamma}$ among $\{X_{i\gamma},\ 1 \le \gamma \le 100\}$ arranged in ascending order of magnitude, where the smallest (least dissimilarity) score is taken as rank 1. In theory, $X_{i\gamma}$ should increase as γ increases (i.e. an upward trend). Therefore, the average value $\bar{A}$ of 5000 A_i scores, with $A_i = \sum_{\gamma=1}^{100} (R_{i\gamma} - \gamma)^2$ being the Spearman's rank statistic for testing an upward trend, is used to compare the performance of dissimilarity measures, where the measure with the smallest $\bar{A}$ favors the upward trend the most and is considered to be most advantageous. The ranks for BLAST are computed by putting the

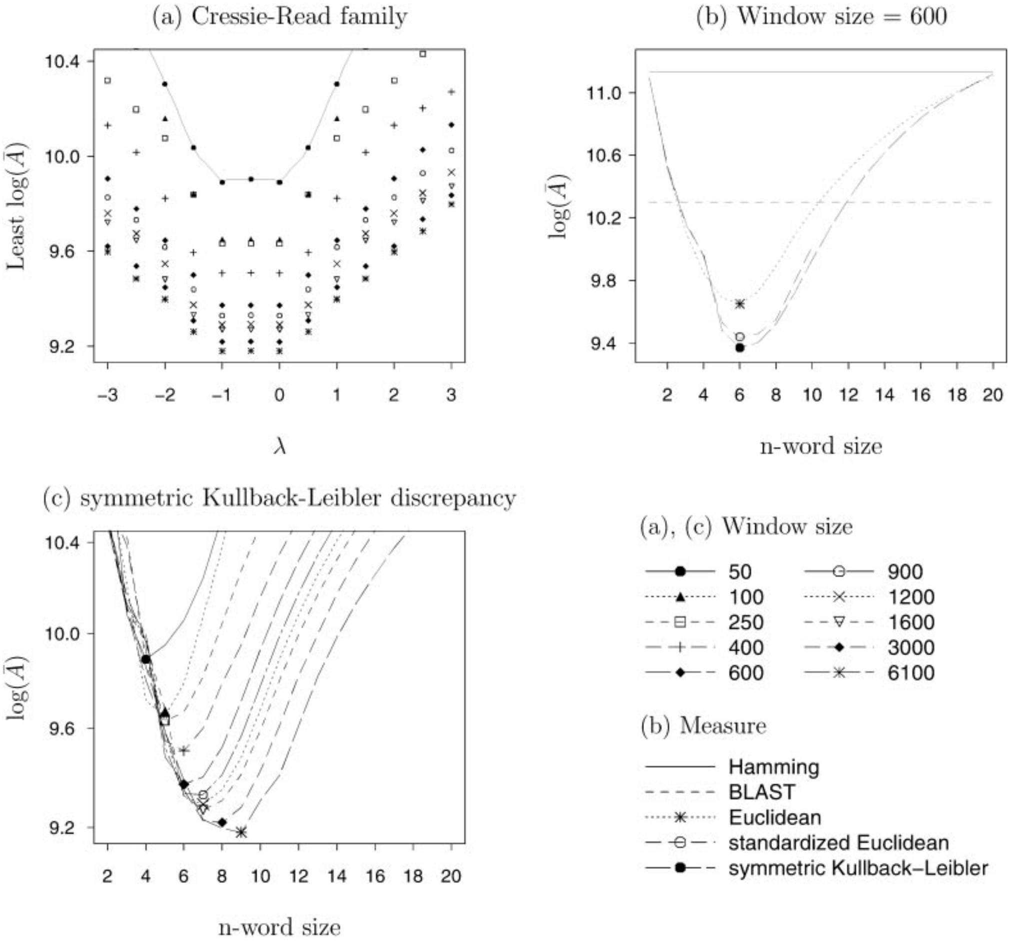

Fig. 2. Over 5000 comparisons: (**a**) the least $\log(\bar{A})$, resulting from using optimal word size, for the word-based measure SC–RD at $\lambda \in [-3,3]$ and different window sizes, where $\bar{A}$ denotes the average score of the Spearman's rank statistic for testing an upward trend; (**b**) $\log(\bar{A})$ among different similarity/dissimilarity measures at window size 600; and (**c**) $\log(\bar{A})$ among different window sizes for the word-based measure I_n. In (b) and (c) the optimal word size associated with that window size is marked. Similar results are obtained for other cases of window sizes in (a)–(c).

100 similarity scores in descending order of magnitude, where the largest (most similarity) score is taken as rank 1. The scores that are <20 (the version BLASTN 2.1.3 is used in our study, see http://www.ncbi.nlm.nih.gov/BLAST, which only provides this bound for very non-similar pairs of sequences) are treated as ties. If several scores are tied, then they are assigned the same rank which is the average of all the tied ranks (called mid-ranks in the area of rank statistics). For an alternative way of comparing the performance of measures, see Remark 3.

Figure 2a shows that at every window size considered (only the results at some selected window sizes are shown for clarity), if the optimal word size, obtained by searching over $1 \leq n \leq 20$, is used, then the performance of SK–LD is the best among the family of SC–RD. Indeed, the curves in Figure 2a are symmetric with respect to $\lambda = -1/2$ and are nearly horizontal over the interval $[-1,0]$ with the value at $\lambda = 0$ a little smaller than those at $-1 < \lambda < 0$. For the rest, we concentrate on the comparison of SK–LD with other type of measures. The comparison of d_n^2, S_n^2, I_n with the Hamming distance and BLAST at window size 600 is given in Figure 2b, where the optimal word size, associated with the smallest $\bar{A}$, is marked. It shows that when the optimal word size is used, the performance of SK–LD is the best, followed by SED, then ED, then BLAST, and Hamming distance is the least favorable. The same performance rankings also hold at all the other window sizes considered. We also find that the optimal word size for each measure increases with window size, as shown in Figure 2c (only the case of SK–LD at some selected window sizes is shown to save space). For example, at window sizes that are multiples of 50 and are (1) in the groups 100–250, 300–700, 750–2500, 2550–4950 and 5000–6100, the optimal word sizes for SK–LD are 5, 6, 7, 8 and 9, respectively; (2) in the groups 100–250, 300–1150, 1200–5100 and 5150–6100, the optimal word sizes for ED are 5, 6, 7 and 8, respectively; and (3) in the groups 100–350, 400–850, 900–1550 and 1600–3000, the optimal word sizes for SED are 5, 6, 7 and 8, respectively (See Remark 4 for a discussion). They also show that for each measure, the smallest $\bar{A}$, resulting from using the optimal word size, decreases with window size. This implies that, for whole sequence similarity/dissimilarity comparison, the window size taken should be as large as possible (but probably not >3000, as restricted by CPU time in practice). A practical implementation is

Table 1. The optimal word size n_l^* of SK–LD for DNA sequence comparison using window size l

l	10–11	12–24	25–70	71–268	269–715	716–2539	2540–4999	5000–6100
n_l^*	2	3	4	5	6	7	8	9

to let the window size be the minimum of 3000 and the lengths of the two DNA sequences under comparison.

For the rest, n_l^* denotes the optimal word size associated with window size l for SK–LD. The scatter plot (not shown to save space) of the pairs (l, n_l^*), $l \in B = \{10, 20, 50, 100 + 50j$: $j = 0, 1, 2, \dots, 120\}$ shows n_l^* increases with l. Moreover, the correlation coefficient of these pairs is 0.871 and that of the pairs $(\log l, n_l^*)$ is 0.957. This provides strong statistical evidence of the monotonic relationship between n_l^* and l. Table 1 give the (interpolated) optimal word size for SK–LD at every window size between 10 and 6100, where the boundaries (i.e. $l = 11$, 12, 24, 25, etc.) are determined by actual simulation (see Remark 5 for a discussion). It is worth noting that the optimal n we recommend for the ED d_n^2 is consistent with the word size used in examples of Torney *et al*. (1990) and Hide *et al*. (1994). See Remark 6 for details.

3.2 Estimation of the degree of dissimilarity

This section describes how to estimate the degree of dissimilarity β between any pair of DNA sequences using the SK–LD I_n. Note that $0 \le \beta \le 1$ with $\beta = 0$ standing for the least dissimilar case and $\beta = 1$ for the most dissimilar case. At each window size $l = 11$, 12, 24, 25,70, 71, 268, 269, 715, 716 (see boundaries between groups in Table 1) and 10, 20, 50 and $100 + 50j, j = 0, 1, 2, \dots, 48$ we generate 5000 I_n scores, $1 \le n \le 20$, resulting from 5000 independent m–s pairs, with the length of every mother being l, at each of the 100 mutation rates $\gamma\%$, $\gamma = 1, 2, \dots, 100$. If we identify the degree of dissimilarity β with the mutation rate, then at each window size l and word size n: (1) the 5000 I_n scores at each mutation rate $\gamma\%$ $(=\beta)$ can be viewed as a random sample from the population $U_{n,\gamma}$ of all I_n scores resulting from the unlimited virtual pool of all m–s pairs of DNA sequences at that mutation rate β and (2) the 500 000 I_n scores, resulting from pooling together the scores for all mutation rates, can be viewed as an approximately random sample from the population $U_n = \cup_{\gamma=1}^{100} U_{n,\gamma}$. Therefore, a β-th quantile of U_n can be estimated by the counterpart of the empirical distribution function associated with the 500 000 I_n scores. Picking $n = n_l^*$ leads to Table A1. Thus, based on the statistic $I_{n_l^*}$, we can use Tables A1 to estimate β. This way of estimating β may be called the pooling method (see Remark 7 for a discussion).

Figure 3 shows the relation between $\hat{\beta}$ and all window sizes (for clarity only the case when window size ≥ 25 is shown) included in Table A1 at some selected SK–LD scores, where the jumps of the curves are due to change of word size and occur at vertical lines at 70.5, 268.5 and 715.5 (as boundaries between word sizes 4 and 5, 5 and 6, and 6 and 7, respectively, see Table 1). Figure 3 shows that for every fixed SK–LD score, $\hat{\beta}$ is monotonically decreasing with window size. Therefore, the method of linear interpolation can be applied to obtain the approximate estimate $\hat{\beta}_l$ of β at any window size l satisfying $l_1 \le l \le l_2$, where both l_1 and l_2 are window sizes that are included in Tables A1 and are closest to l. Although the true relationship between the estimate and window size may not be linear, the linear interpolation should provide a good approximation, especially when $l_2 - l_1$ is small. Let $\hat{\beta}_j$ denote the estimate of β by matching the observed SK–LD score $I_{n_l^*}$ against Table A1 with window size l_j and optimal word size $n_{l_j}^*$ (note that we have designed Table A1 so that $n_{l_1}^* = n_{l_2}^* = n_l^*$ and $l_2 - l_1 \le 50$ always). Then $\hat{\beta}_l$ is between $\hat{\beta}_1$ and $\hat{\beta}_2$. Thus, $\hat{\beta}_l = a\hat{\beta}_1 + (1 - a)\hat{\beta}_2$, where $a = (l_2 - l)/(l_2 - l_1)$. A numerical example is given in Section 4.2.

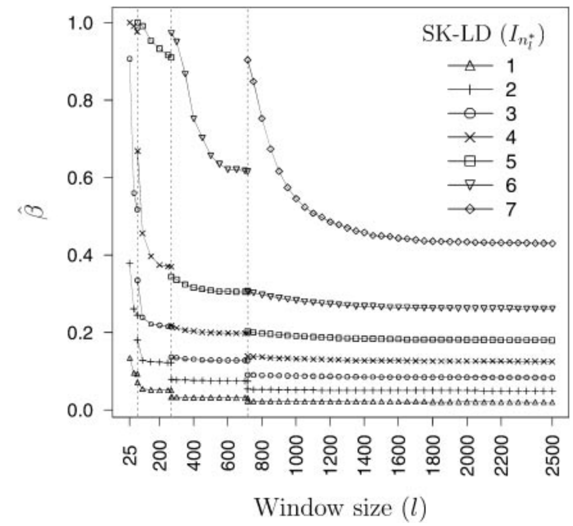

Fig. 3. Relation between $\hat{\beta}$ and all window sizes (only the case when window size ≥ 25 is shown for clarity) included in Table A1 for $I_{n_l^*} = 1, \dots, 7$. The jumps of the curves are due to changes of word size $n = n_l^*$ and occur at the vertical lines at 70.5, 268.5 and 715.5 (as the boundaries between word sizes 4 and 5, 5 and 6, and 6 and 7, respectively, see Table 1). Note that approximately $I_4 \le 4.864$ for $25 \le l \le 70$, $I_5 \le 5.737$ for $71 \le l \le 268$, $I_6 \le 6.693$ for $269 \le l \le 715$ and $I_7 \le 7.712$ for $716 \le l \le 2500$. Results for other cases of $I_{n_l^*}$ are similar.

4 EXPERIMENTAL RESULTS

Throughout the data analysis, for any pair of DNA sequences under comparison, the window size is taken to be the minimum of their lengths for Experiments #1–#3 and the probe length for Experiment #4, and the SK–LD is computed at the optimal word size (Table 1), which leads to $\hat{\beta}$ via Table A1. Since the word size may vary with the pair of sequences under comparison, the present setup is more efficient than the one in Wu *et al*. (1997, 2001) in which the word size is not optimally chosen and does not vary with window size. For base compositions of all the DNA sequences used in our experiments (which vary from U-I, skewed to Markov chain of order up to 5), see Remarks 8–11.

Table 2. Score matrix among *thr*A-*thr*C using dissimilarity measure SK–LD (upper triangle) and similarity measure BLAST (lower triangle) at the default search parameter setting

	ec-thrA	*ec*-thrB	*ec*-thrC	*sf*-thrA	*sf*-thrB	*sf*-thrC	*rand*-thrA
ec-thrA		0.4294	0.3801	0.0249	0.4461	0.3733	0.5470
ec-thrB	26.26		0.4507	0.4124	0.0245	0.4435	0.8025
ec-thrC	<20	22.30		0.3982	0.4662	0.0238	0.6387
sf-thrA	4629.30	26.26	<20		0.4235	0.3888	0.5636
sf-thrB	26.26	1731.10	22.30	26.26		0.4589	0.8716
sf-thrC	<20	22.30	2464.60	<20	22.30		0.6378
rand-thrA	<20	24.28	22.30	<20	22.30	22.30	

Note: The diagonal entries are all 0's for SK–LD, and 4883, 1850, 2551.8, 4883, 1850, 2551.8 and 4883 for BLAST.

4.1 Experiment #1

Six DNA sequences are taken from the threonine operons of *Escherichia coli* K-12 (gi:1786181) and *Shigella flexneri* (gi:30039813) of Genbank. The three sequences taken from each threonine operons are *thr*A (aspartokinase I-homoserine dehydrogenase I, 2463 bp), *thr*B (homoserine kinase, 933 bp) and *thr*C (threonine synthase, 1287 bp), using the ORF's 337–2799 (*ec*-thrA), 2801–3733 (*ec*-thrB) and 3734–5020 (*ec*-thrC) in the case of *E.coli* K-12, and using 336–2798 (*sf*-thrA), 2800–3732 (*sf*-thrB) and 3733–5019 (*sf*-thrC) in the case of *S.flexneri*. In addition, a sequence (*rand*-thrA) is randomly generated according to the base probabilities and length of *ec*-thrA for comparison.

The estimated degree of dissimilarity $\hat{\beta}$ using SK–LD and similarity scores using BLAST, at the default (search) parameter setting, between the 7 sequences are shown in Table 2. BLAST scores at many other parameter settings have also been obtained but not shown in Table 2 to save space. The results obtained by $\hat{\beta}$ agree with those by the chaos game representation (Almeida *et al*., 2001), by PSM of Pham and Zuegg (2004) and by BLAST at a few parameter settings (e.g., the setting: -G 5 -E 2 -q -2 -r 3 -W 8), in which the thrA sequences are closer to thrC than to thrB, and thrB closer to thrA than to thrC. However, for BLAST, the result obtained at most parameter settings is the same as that at the default parameter setting which shows a slightly different relationship between the three sequences. Specifically, it put the thrB sequences closer to thrA than to thrC, and thrC closer to thrB than to thrA. The difference is also illustrated by the dendrogram in Figure 4. Moreover, the dendrogram shows that all the real DNA sequences are more closely clustered using $\hat{\beta}$ than using BLAST at the default setting. This suggests that, at distinguishing a randomized sequence from a group of related real DNA sequences as a whole, $\hat{\beta}$ is no worse than BLAST at all parameter settings and better than BLAST at most parameter settings.

4.2 Experiment #2

Both $\hat{\beta}$ (or equivalently, SK–LD) and BLAST are used to perform a search for dissimilarities/similarities of the query sequence HSLIPAS (1612 bp) human lipoprotein lipase against a test dataset of 63 library sequences chosen from many different divisions of Genbank and vary from mammals, invertebrates, viruses, plants, bacteria, etc. [see Table A2 at http://www.stat.ncku.edu.tw/tjwu. It expands both Table 2 of Hide *et al*. (1994) and Table 1 of Wu *et al*. 1997)]. The 63 sequences of the test dataset vary in length from 322 to 2 462 499 bases. Every member of the test dataset is classified as being related or not related in biological function to the query sequence. There are 35 sequences classified as being related (they are numbered from 1 to 35 herein), and 28 sequences classified as being not related (they are numbered from 36 to 63 herein).

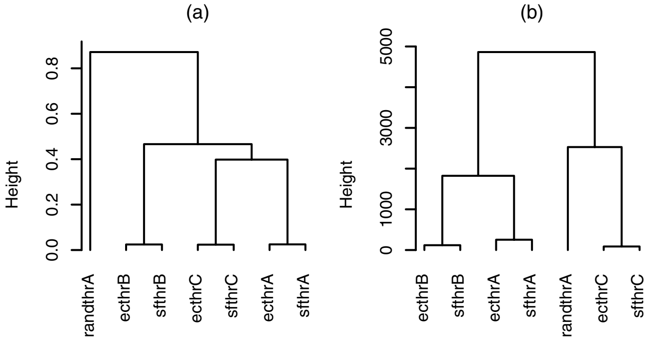

Fig. 4. Hierarchical dendrogram (using complete linkage) of *thr*A, *thr*B, *thr*C and *rand* sequences on the basis of the matrices of similarity/dissimilarity scores in Table 2, using (**a**) $\hat{\beta}$, (**b**) BLAST at the default search parameter setting.

The $\hat{\beta}$ and BLAST scores between HSLIPAS and 63 library sequences are sorted from the highest to lowest similarity, respectively, and the sensitivity and selectivity are used to quantify their performances. Sensitivity is defined to be the number of HSLIPAS-related sequences found among the first 35 library sequences. Selectivity is measured in terms of consecutive correct classifications, which means, starting from the first sequence, the total number of sequences are counted until the first non-HSLIPAS-related library sequence occurs. Thus, a score of 35 for sensitivity and selectivity is a perfect score for this collection of sequences.

We find the sensitivity and selectivity for $\hat{\beta}$ are 34 and 30, respectively, and those for BLAST are 29 and 22, respectively, at the default parameter setting and are no better than 33 and 28, respectively, at other parameter settings (the optimal result is obtained, e.g. at: -G 5 -E 2 -q -1 -r 1 -W 7). Hence, $\hat{\beta}$ performs better than BLAST. Also, $\hat{\beta}$ improves the combined K-LD (Wu *et al*., 2001), whose sensitivity and selectivity are 31 and 24, respectively. Finally, the sensitivity and selectivity of PSM of

Table 3. Comparison of average rank of BLAST scores of probes to non-target genes in designing a 70mer oligo probe for each gene from T7 phage genome

BLAST parameter	Method			BLAST parameter	Method		
	$\hat{\beta}$	OP	YODA		$\hat{\beta}$	OP	YODA
Default	53.93	76.86	76.37	-G 5 -E 3 -q -2 -r 3 -W 8	60.50	65.42	81.34
-G 5 -E 2 -q -1 -r 1 -W 8	59.91	64.86	82.52	-G 5 -E 3 -q -3 -r 2 -W 11	53.43	74.60	79.19
-G 5 -E 2 -q -1 -r 2 -W 9	64.14	72.28	70.61	-G 5 -E 3 -q -2 -r 1 -W 10	51.71	76.12	79.40
-G 5 -E 2 -q -2 -r 1 -W 7	53.13	74.43	79.67	-G 6 -E 2 -q -2 -r 1 -W 11	53.88	75.25	78.07
-G 5 -E 2 -q -2 -r 3 -W 11	52.39	74.83	80.02	-G 6 -E 2 -q -2 -r 1 -W 12	56.40	75.18	75.56

NOTE: OP stands for OligoPicker. Default: -G 5 -E 2 -q -3 -r 1 -W 11. The BLAST score of a probe equals the largest BLAST score between that probe and all non-target genes. We combine all 137 ($= 46 + 46 + 45$) BLAST scores of probes and rank them in ascending order of magnitude (use mid-rank for tied scores, see Section 3.1), where the *i*-th smallest BLAST score is taken as rank *i*. Each entry is the average rank of a method.

Pham and Zuegg (2004) are 32 and 26, respectively. Thus, $\hat{\beta}$ performs more favorably than PSM.

As a numerical example to explain how to use Tables A1 to estimate β, we compare the library SSLPLRNA (2963 bp) and the query HSLIPAS. We take $l = 1612$ and $n_l^* = 7$ (Table 1) to obtain $I_{n_l^*} = 4.2294$. Since $l_1 < l < l_2$ with $l_1 = 1600$ and $l_2 = 1650$ and $a = (1650 - 1612)/(1650 - 1600) = 0.76$, it follows by linear interpolation (see Section 3.2) and Table A1 that $\hat{\beta} = (0.76)(0.1382) + (0.24)(0.1381) = 0.13818$.

It is worth mentioning that, using a PC with Pentium 4 processor running at 3.4 GHz CPU and 1 GB RAM, it takes only ~4.4 CPU seconds to finish the computation of SK–LD between the query and all the 59 library sequences: #1–#31 and #36–#63 (varying from 322 to 22 257 bp and average = 3845 bp in length, see Table A2), and 189 CPU seconds between the query and all the 4 library sequences: #32–#35 (varying from 1 173 390 to 2 462 499 bp and average = 1 838 935 bp in length). Here the total memory space required, in addition to that for running the MATLAB program, varies from 4 to 22 MB over all library sequences. Therefore, our algorithm is fairly efficient.

4.3 Experiment #3

A protein is translated from a gene composed of several exons in the DNA sequence. Most likely, some of the exons are shuffled during evolution. Human and mouse genomes are not far away in the evolution tree and share many common segments. However, segments in a chromosome of mouse might have their similar segments in more than one chromosome of human. This is an example of DNA subsequences shuffling during evolution.

Fifteen severe acute respiratory syndrome ORF sequences, varying in length from 231 to 8628 bases, are chosen from Genbank. The names of their loci are *AY*707461, *AY*536760, *AY*536759, *AY*536758, *AY*702026, *AY*648300, *AY*569693, *AY*365036, *AY*525636, *AY*444813, *AY*322205S4, *AY*451866, *AY*707854, *AY*609081, *AY*322205S3 (labelled herein as $g1, \dots, g15$). We choose *g1* (1605 bp) as the query sequence and cut it into three segments of equal lengths, by permuting them, we obtain five new sequences $g1_1, \dots, g1_5$. Let *T(a,b)* denote the similarity/dissimilarity score between sequences *a* and *b* using the measure *T*. We compare the ratio T($g1$,gi)/T($g1_j$,gi) for T = SK–LD or BLAST. We find that for the five groups ($j = 1, \dots, 5$), while each group contains 14 ratios ($i = 2, \dots, 15$), the group mean varies from 0.944 to 0.97 and SD from 0.057 to 0.088 for SK–LD, whereas the group mean varies from 1.244 to 1.978 and SD from 0.261 to 1.045 for BLAST at the default parameter setting (similar results are obtained at other parameter settings). Therefore, SK–LD is much less sensitive than BLAST to segment shuffling. Such a property of SK–LD is desirable in reconstructing the evolution tree because shuffled DNA sequences (if no other biological factor is present to affect the changes) should not be far away from one another in the tree.

4.4 Experiment #4

In Experiments #1−#3, we focus on whole sequence similiarity/dissimilarity identification. However, in many applications it is the local similarity that matters the most. As a demonstration of the applicability of our method in local similarity comparison situations, we choose the T7 phage genome (39 937 bp), which includes 60 genes with lengths varying from 90 to 3957 bp, as the test dataset to show how $\hat{\beta}$ can help in selecting oligo probes for use in gene expression microarray design. For simplicity, we focus on the selection of a single 70mer oligo probe for each gene.

Pick any gene, say G_0, and any window W_0 of size $l = 70$ bp from G_0. The SK–LD between the window W_0 and the remaining set $\mathcal{F}$ of 59 genes, with optimal word size $n_l^* = 4$ (Table 1), is defined by $I_4(W_0, \mathcal{F}) = \min_{G \in \mathcal{F}} I_4(W_0, G)$ where $I_4(W_0, G)$ is the SK–LD between W_0 and G (recalling Section 2). The window that maximizes $I_4(W_0, \mathcal{F})$ over all W_0 is taken as a probe for G_0 and its corresponding $\hat{\beta}$ value can be obtained from Table A1. By rejecting 13 redundant genes (with $\hat{\beta} = 0$) and a gene that is excessively similar to non-target genes (with $\hat{\beta} \leq 0.25$; threshold value other than 0.25 may be used), we obtain a probe for each of the remaining 46 genes. Their loci names, probes and $\hat{\beta}$ can be found in Table A3 at http://www.stat.ncku.edu.tw/tjwu.

The above approach does not use BLAST at all. It uses $\hat{\beta}$ along with a threshold value set by the user to avoid probes with excessive sequence similarity to a non-target gene that may be present during the hybridization. Thus it should increase the specificity of individual probes. A probe must satisfy three conditions (Hughes *et al.*, 2001; Kane *et al.*, 2000; Nordberg, 2005) to be specific: (1) total percent identity must be ≤75–80% with a non-target gene; (2) it must not include a stretch of identical sequence >15 contiguous bases with a non-target gene; (3) it must not include any low complexity region (e.g. long stretches of the same base, homopolymeric runs, etc.).

We compare our method with OP (OligoPicker, Wang and Seed, 2003) and YODA (Nordberg, 2005). OP also selects 46 probes corresponding to the same 46 genes as ours, while YODA only selects 45 probes corresponding to 45 genes contained in ours

(YODA rejects gene T7p25). All the 137 (= 46 + 46 + 45) probes satisfy the specificity conditions (1)–(3), as can be seen by using the Hamming distance, BLAST and DUST programs (Hancock and Armstrong, 1994). For example, using Hamming distance the average percent identity and SD of probes with non-target genes selected by $\hat{\beta}$, OP and YODA are 49.16 and 2.3%, 48.82 and 2.5 and 50.19% and 3%, respectively, and there is little difference among them. Next, let the BLAST score of a probe equal the largest BLAST score between that probe and all non-target genes, and then combine all such 137 BLAST scores and rank them in ascending order of magnitude (recalling Section 3.1). Evidently, a smaller average rank corresponds to less similarity, and hence higher specificity. Table 3 shows probes selected by $\hat{\beta}$ has much smaller average rank than those by OP and YODA. In conclusion, our study, although very primitive, shows $\hat{\beta}$ can help in locating probes with high specificity and has potential for probe design.

5 DISCUSSION

After computing a similarity/dissimilarity score between any pair of DNA sequences, we are sometimes not certain about the relative magnitude of the score since the terms ‘small score’, ‘moderate score’, and ‘large score’ are relative terms. Consequently, the term ‘dissimilar sequences’ is not clearly defined using these scores. This motivates us to introduce in Section 3.2 the so-called ‘degree of dissimilarity’ β between any pair of DNA sequences and develop a method to estimate the degree of dissimilarity based on SK–LD score. In this way a clear cut definition of ‘dissimilar sequences’ is made possible through the estimated values $\hat{\beta}$ of β, by $\hat{\beta} > \gamma\%$ for some γ. The exact choice of γ depends on the user’s objective. For example, if $\hat{\beta}$ is used as a filter in the database search, then small γ results in filtering out a large number of dissimilar sequences, and thus dramatically speeds the database search for similar sequences. The subjects of future research are (1) the expansion of Table A1 to include larger window sizes (currently it only includes results up to window size 2500 due to restriction on our computing environment), (2) the extension of our method to (hidden) Markov chain model of base composition and other types of mutation, (3) the extensive investigation of the specificity, along with sensitivity and consistency [see, e.g. Nordberg (2005) for definitions], of single or multiple oligo probes selected by $\hat{\beta}$ for use in microarray design, (4) the enhancement of our simple consecutive base sliding word model to incorporate frame shifts (e.g. using not just one but three sliding words each picking up 1 in every 3 bases and can be computed independently for statistical characteristics), and/or to use nonconsecutive base word models similar to those presented in Huang *et al*. (2004) and (v) the use of our method in protein sequence comparisons (specifically, the use of our method in producing a better amino acid matching table such as the BLOSUM or PAM matrices).

ACKNOWLEDGEMENTS

The authors would like to express their sincere thanks to two referees for many valuable suggestions that significantly improved the quality and presentation of the article. Part of the research was done while T.-J.W. was visiting the Institute of Statistical Science, Academia Sinica, Taipei, Taiwan.

Conflict of Interest: none declared.

REFERENCES

Alberts,B., Bray,D., Lewis,J., Raff,M., Roberts,K. and Watson,J.D. (1994) *Molecular Biology of the Cell*. 3rd ed. Garland, New York.

Almeida,J.S. *et al*. (2001) Analysis of genomic sequences by chaos game representation. *Bioinformatics*, **17**, 429–437.

Altschul,S.F. *et al*. (1990) Basic local alignment search tool. *J. Mol. Biol*., **215**, 403–410.

Altschul,S.F. *et al*. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*., **25**, 3389–3402.

Arratia,R. *et al*. The Erdös-Rényi law in distribution, for coin tossing and sequence matching. *Ann. Stat*., **18**, 539–570.

Blaisdell,B.E. (1986) A measure of the similarity of sets of sequences not requiring sequence alignment. *Proc. Natl Acad. Sci. USA*, **83**, 5155–5159.

Blaisdell,B.E. (1989a) Effectiveness of measures requiring and not requiring prior sequence alignment of estimating the dissimilarity of natural sequences. *J. Mol. Evol*., **29**, 526–537.

Blaisdell,B.E. (1989b) Average value of a dissimilarity measure not requiring sequence alignment are twice the averages of conventional mismatch count requiring sequence alignment for a computer-generated model system. *J. Mol. Evol*., **29**, 538–549.

Cressie,N. and Read,T.R.C. (1984) Multinomial goodness-of-fit tests.. *J. R. Stat. Soc. Ser. B*, **46**, 440–464.

Fichant,G. and Gautier,C. (1987) Statistical method for predicting protein coding regions in nucleic acid sequences. *CABIOS*, **3**, 287–295.

Frith,M.C. *et al*. (2004) Finding functional sequence elements by multiple local alignment. *Nucleic Acids Res*., **32**, 189–200.

Gentleman,J.F. and Mullin,R.C. (1989) The distribution of the frequency of occurrence of nucleotide subsequences, based on their overlap capability. *Biometrics*, **45**, 35–52.

Hancock,J.M. and Armstrong,J.S. (1994) SIMPLE34: an improved and enhanced implementation for VAX and Sun computers of the SIMPLE algorithm for analysis of clustered repetitive motifs in nucleotide sequences. *Comput. Appl. Biosci*., **10**, 67–70.

Hide,W. *et al*. (1994) Biological evaluation of d^2, an algorithm for high performance sequence comparison. *J. Computat. Biol*., **1**, 199–215.

Huang,X. *et al*. (2004) Efficient combination of multiple word models for improved sequence comparison. *Bioinformatics*, **20**, 2529–2533.

Hughes,T.R. *et al*. (2001) Expression profiling using microarrays fabricated by an ink-jet oligonucleotide synthesizer. *Nat. Biotechnol*., **19**, 342–347.

Kane,M.D. *et al*. (2000) Assessment of the sensitivity and specificity of oligonucleotide (50mer) microarrays. *Nucleic Acids Res*., **28**, 4552–4557.

Nordberg,E.K. (2005) YODA: selecting signature oligonucleotides. *Bioinformatics*, **21**, 1365–1370.

Pearson,W.R. (1990) Rapid and sensitive sequence comparison with FASTA and FASTP. *Methods Enzymol*., **183**, 63–98.

Pearson,W.R. and Lipman,D.J. (1988) Improved tools for biological sequence comparison. *Proc. Natl Acad. Sci. USA*, **85**, 2444–2448.

Pevzner,P.A. (1992a) Nucleotide sequences versus Markov models. *Comput. Chem*., **16**, 103–106.

Pevzner,P.A. (1992b) Statistical distance between texts and filtration methods in sequence comparison.. *CABIOS*, **8**, 121–127.

Pham,T.D. and Zuegg,J. (2004) A probabilistic measure for alignment-free sequence comparison. *Bioinformatics*, **20**, 3455–3461.

Pinheiro,H.P., Moiseiwitsch,F.S. and Sen,P.K. (2000) Analysis of variance based on the hamming distance. In Sen,P.K. and Rao,C.R. (eds), *Handbook of Statistics Volume 18: Bioenvironmental and Public Health Statistics*. Oxford, North-Holland, p. 735.

Sege,R.D. and Saxberg,B.E.H. (1982) A statistical test for comparing several nucleotide sequences. *Nucleic Acids Res*., **10**, 375–389.

Torney,D.C., Burks,C., Davison,D. and Sirkin,K.M. (1990) Computation of d^2: a measure of sequence dissimilarity. In Bell,G. and Mrarr,T. (eds), *Computers and DNA, Santa Fe Institute Studies in the Sciences of Complexity*. Addison-Wesley, New York, pp. 109–125.

Vinga,S. and Almeida,J. (2003) Alignment-free sequence comparison-a review. *Bioinformatics*, **19**, 513–523.

Wang,X. and Seed,B. (2003) Selection of oligonucleotide probes for protein coding sequences. *Bioinformatics*, **19**, 796–802.

Waterman,M.S. (ed.), *Mathematical Methods for DNA Sequences*. CRC, Boca Raton, FL.

Wu,T.-J. *et al*. (1997) A measure of DNA sequence dissimilarity based on Mahalanobis distance between frequencies of words. *Biometrics*, **53**, 1431–1439.

Wu,T.-J. *et al*. (2001) Statistical measures of DNA sequences dissimilarity under Markov chain models of base composition. *Biometrics*, **57**, 441–448.