

SEPA: Approximate Non-Subjective Empirical p -Value Estimation for Nucleotide Sequence Alignment

Ofer Gill and Bud Mishra

Courant Institute of Mathematical Sciences,
New York University, 251 Mercer Street, New York NY 10012, USA,
gill@cs.nyu.edu,
<http://bioinformatics.nyu.edu/~gill/index.shtml>

Abstract. In the bioinformatics literature, pairwise sequence alignment methods appear with many variations and diverse applications. With this abundance, comes not only an emphasis on speed and memory efficiency, but also a need for assigning confidence to the computed alignments through p -value estimation, especially for important segment pairs within an alignment. This paper examines an empirical technique, called SEPA, for approximate p -value estimation based on statistically large number of observations over randomly generated sequences. Our empirical studies show that the technique remains effective in identifying biological correlations even in sequences of low similarities and large expected gaps, and the experimental results shown here point to many interesting insights and features.

1 Introduction

In the field of comparative genomics, an emphasis is placed on its functional genomics aspects. Most often we align two or more sequences, because we expect that the important areas selected from that alignment will point to a significant common biological function, even when we realize that there can be no absolute guarantee of this. In order to draw our attention very quickly to the most pertinent similar subsequences, it is necessary to compare the important areas of alignments and rank them in order of their relevance. For instance, by comparing alignments in related sequences to those of unrelated sequences with no common biological function, we may derive, for any alignment, the probability that its important areas occur by mere coincidence. This probability measure is also known as a p -value, and low p -values relate to high relevance rank.

Many p -value estimation techniques have been suggested and examined previously, for instance, Karlin-Altschul [7] and Siegmund-Yakir [14], but none have proven completely satisfactory. In this paper, we focus on using empirical results to improve the p -value approximation in case of alignments of noncoding nucleotide sequences of lengths varying from .5 Kb to 12 Kb, with expected large gaps and low similarities. These alignments are often computed with the complex but biologically faithful model involving piecewise-linear gap penalty functions as in PLAINS [3]; nonetheless, other techniques such as LAGAN, EMBOSS, and LALIGN have also proven effective. We demonstrate the effectiveness of a p -value approximation technique called SEPA (Segment Evaluator for Pairwise Alignments) as it selects and scores important segments pairs. Furthermore, for random sequences, we also empirically characterize how various alignment statistics, such as the segment pair lengths, scores, and magnitudes, distribute as a function of sequence lengths. From this analysis, the parameters for a p -value approximation are estimated, and used to demonstrate the method of sensitivity in distinguishing important homologies from unimportant chance occurrences of subalignments within sequences. Furthermore, SEPA is non-subjective, since it can easily be applied to any alignment tool. We will illustrate this advantage by using it to compare the results of PLAINS with LAGAN, EMBOSS, and LALIGN. Because of these strengths and despite its empirical foundation, SEPA fulfills a practical computational need by speeding up the core search processes in comparative genomics.

2 Overview

We introduce some notations as follows: Assume the sequences to be aligned are X and Y , and their respective lengths are m and n , where $m \geq n$. Let X_u and Y_v denote respectively the u^{th} character of X and the v^{th} character of Y , where $1 \leq u \leq m$ and $1 \leq v \leq n$.

Let us suppose that aligning X and Y with some arbitrary alignment tool produces an alignment A of length a , where $m \leq a \leq m + n$. We will represent an alignment A as follows: For each i , $A[i]$ denotes the i^{th} position in alignment A , and it is represented as a pair of index coordinates (u, v) taken from X and Y , and this corresponds to X_u and Y_v being aligned to each other at position i in A if $u > 0$ and $v > 0$, or one of X_u or Y_v being aligned against a gap if either $v \leq 0$ or $u \leq 0$.

Next, let $A[i : j]$ denote the portion of alignment $A[i], A[i + 1], \dots, A[j]$. We will refer to $A[i : j]$ as a *strip* or *segment pair* from position i to position j .

Let $ww(i)$ denote the penalty for a gap of length i . $ww(\cdot)$ can be any arbitrary function, but for this paper, we will assume it is a p -part piecewise-linear function where each successive slope is smaller than the previous one. A more specific version of this score-function is where $p = 1$, which is the affine function used in the Smith-Watermann algorithm.

Also, let $S(i, j)$ denote the score for strip $A[i : j]$ where the score is computed by adding following values: m_a is a score for each match, m_s is the penalty for each mismatch, and $ww(\cdot)$ is used to penalize the gaps. To compute $S(i, j)$ from $A[i : j]$, each match and mismatch within it is added or deducted from the score individually, while each region of X against a gap and Y against a gap is penalized as a whole using $ww(\cdot)$ based on the length of that region.

Suppose we have a scheme that marks r non-overlapping strips as important. Suppose that the endpoints for these strips are denoted as $(i_1, j_1), (i_2, j_2), \dots, (i_r, j_r)$. For each k , we wish to measure in some way how strip $A[i_k : j_k]$ provides a meaningful correlation between X and Y . One common mathematical approach is to, given a certain null hypothesis, compute the p -value of $Pr(x \geq s)$ where $s = S(i_k, j_k)$. This p -value is known as the coincidental probability of obtaining a strip with score at least s . For this paper, we will assume the null-hypothesis is the behavior of important strips taken from pairwise-aligning randomly generated DNA sequences. Also, if the total scores of all strips is $t = \sum_{k=1}^r S(i_k, j_k)$, then $\zeta = Pr(x \geq t, y \leq r)$, the probability of obtaining at least a total score of t using at most r strips.

One should note that coincidental probabilities of the segments (both p -values and ζ) are dictated by the scheme used to determine the segments as important. One scheme might deem strip $A[i : j]$ as important, but SEPA might not, and instead SEPA may consider a possibly overlapping strip $A[i' : j']$ as important. As a result, the formula for the p -values and ζ value could differ from one scheme to the other. For instance, in the method used to obtain important segments mentioned in Karlin-Altschul [7], $Pr(x \geq s) = 1 - \exp(Kmne^{-\lambda s})$ holds. However, as argued later in this paper, for the way SEPA obtains the segments from an alignment A , we approximate the p -value as $Pr(x \geq s) = \frac{K}{\lambda} e^{-\lambda s}$.

2.1 Obtaining High-Scoring Strips from an Alignment

Given an alignment A produced from sequences X and Y , we produce important strips as follows: Given fixed constants W and ω , and ρ (where W is an integer, and ω and ρ are real numbers in the range $[0, 1]$), let W denote the window size to be used, ω denote the value used to prevent portions of A of lowest match percentage from becoming considered as important strips, and ρ denote the value used to filter away areas of A that have too low of a p -value. We obtain our segment pairs in the following steps:

(1) For all i from 1 to $a - (W - 1)$, we compute $p_a(i)$, the percentage of entries in $A[i : i + W - 1]$ where a match has occurred. Let μ and σ denote the mean and standard deviation of our $p_a(\cdot)$ values. Next, for each i , we mark¹ $p_a(i)$ values as ‘‘special’’ if they exceed a threshold value of $\mu + \omega\sigma$. Hence, we filter away $A[i : i + W - 1]$ if it fails to meet this threshold value.

¹ The choice of using $\mu + \omega\sigma$ as the cutoff value instead of a fixed constant gives us the flexibility of catching important regions in the two sequences, regardless of how homologous they are to each other.

(2) For each u and u' (with $u \leq u'$), if $p_a(u), p_a(u+1), \dots, p_a(u')$ are all marked as “special”, but $p_a(u-1)$ and $p_a(u'+1)$ are not, then we consider the strip $A[u : u' + W - 1]$ as important (i.e., we consider as important the strip starting the leftmost entry represented by $p_a(u)$, up till the rightmost entry represented by $p_a(u')$).

(3) For each strip $A[i : j]$ deemed important, we trim it so that it starts and ends at a position in the alignment where a match occurred. Thus, if i' is the smallest value such that $i' \geq i$ and $A[i']$ is a match position, and j' is the largest value such that $j' \leq j$ and $A[j']$ is a match position, then we trim strip $A[i : j]$ into strip $A[i' : j']$.

(4) Next, we merge together any important strips that overlap. Namely, if we have two strips $A[i : j]$ and $A[k : l]$ such that $i \leq k \leq j$, then we merge these strips into one larger strip $A[i : \max(j, l)]$.

(5) With all strips now representing non-overlapping regions, we then proceed to give each strip $A[i : j]$ its corresponding score $S(i, j)$, as well as its p -value. We delete $A[i : j]$ if its p -value exceed ρ , since that indicates that $A[i : j]$ may be coincidental. We can optionally also collect other information at this point, such as the length of each strip.

(6) The r strips kept at this step are considered the “good” ones. We now compute t , the sum of the scores of the these strips. Using this value, we can compute ζ , coincidental probability for all r strips obtained.

Note that these steps for SEPA are similar to that of [3], except that the calculation for each segment pair’s coincidental probability differs. Based on empirical experimentation, setting $W = 50$, $\omega = 0.5$, and $\rho = 0.5$ yields segment pairs that are reasonably long, non-coincidental, and have significantly higher matches than the alignment “background”. We reasoned that since our method of obtaining segment pairs differs from that of Karlin-Altschul, then the method for computing p -values for each segment pair cannot build upon their assumptions.

2.2 Methods: Analyzing Segment Pairs

In order to approximate an appropriate p -value estimation for SEPA, we analyzed segment pairs behavior over our assumed null hypothesis of alignments for randomly generated nucleotide sequences. For length values ranging from 1000 bp to 8000 bp, we generated 25 random sequences. We also generated 25 random sequences of length 500 bp. For each combination of these length pairs, we ran all 625 possible pairwise alignments using PLAINS, and analyzed results using SEPA where $\rho = 1$ (to avoid filtering any segments out due to low p -value). The results for mean length-to-score and mean segment scores are shown in Fig. 1. From this, we infer that both are uniform in terms of m and n . In the appendix, Figures 5 and 6 elaborate further.

For our random sequences, we also observed the average and variance behaviors for r and t in terms of m and n , where r is the number of segment pairs observed, and t is the total score of all the segment pairs. Furthermore we found that the mean for r , variance for r , and mean for t all scale roughly to $k_0 \ln(k_1 mn + k_2(m+n) + k_3)$, and the deviation for t scales roughly to $\max(k_0, k_1 i \cdot d + k_2 i + k_3 d + k_4)$, where $i = \min(m, n)$, $d = \|m - n\|$, and k_0, k_1, k_2, k_3, k_4 are constants². Figures 7 and 8 in the appendix illustrate further how all of this was derived.

Since the average ratio of segment lengths to score is almost uniform in these plots, it suggests that the gap penalty used to score the strips can be treated as if it is a differently-weighted mismatch. Also, note that the p -values computed with the model studied by Siegmund-Yakir[14] differs mildly from the model using the simplifying assumption that gaps are differently-weighted mismatches. For this reason, it is common for tools to ignore the effects of gaps in generating their p -values, much like BLAST³. Thus, we may similarly treat our piecewise-linear gap penalty $ww(\cdot)$ as differently-weighted mismatches in approximating the p -value. Fig. 2 shows a plot of segment scores to frequency from

² For average r , $k_0 = 10^3$, $k_1 = 7.95 \times 10^{-10}$, $k_2 = 1.54 \times 10^{-7}$, $k_3 = 1.01$. For variance of r , $k_0 = 10^3$, $k_1 = 1.93 \times 10^{-10}$, $k_2 = 1.97 \times 10^{-7}$, $k_3 = 1.00$. For average t , $k_0 = 10^5$, $k_1 = 4.29 \times 10^{-10}$, $k_2 = 1.33 \times 10^{-8}$, and $k_3 = 1.00$. For deviation of t , $k_0 = 100$, $k_1 = -5.54 \times 10^{-5}$, $k_2 = 4.63 \times 10^{-1}$, $k_3 = 1.04 \times 10^{-2}$, and $k_4 = -65.01$.

³ The main reason we did not use BLAST in comparing alignment results is because BLAST was unable to align most of the sequences mentioned in table 1.

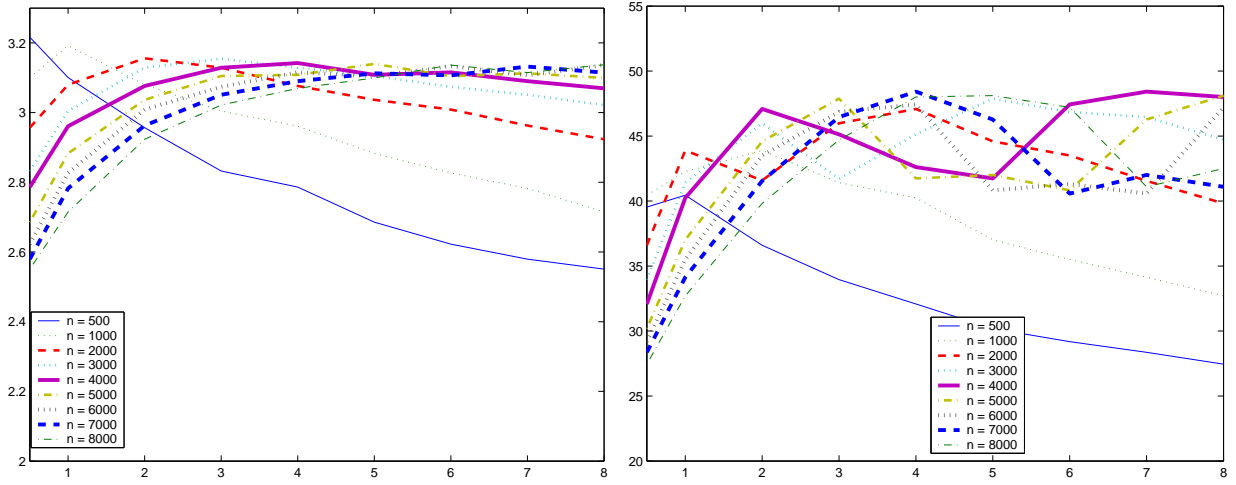


Fig. 1. Shown above are the mean length-to-score ratio and mean segment scores observed in the strips from aligning randomly generated DNA sequences. In the plots shown above, a unique line is plotted corresponding to each value of n in the thousand lengths ranging from 1000 to 8000. For these plots, x represents the m value divided by 1000, and y represents the mean observed for that particular m and n , and the left plots illustrate mean length-to-score ratio for the segment pairs, while the right plots illustrate mean segment pair scores. These plots indicate that, for small n values, the average length-to-score ratio and average score decrease with increasing m . However, asymptotically (for large n) the average length-to-score ratio and average segment scores stay roughly constant in terms of m (at 3.1 and 45 respectively) and don't stray too far. This leads us to infer that length-to-score ratio can be well-approximated by a constant, and that segment scores are independent of m and n .

which we derive our p -value approximation. Using it, we approximate that $P(x = s) = Ke^{-\lambda s}$, with $K = 8.69 \times 10^{-2}$ and $\lambda = 3.26 \times 10^{-2}$. Our p -value of $P(x \geq s)$ is therefore:

$$P(x \geq s) = \int_s^{\infty} Ke^{-\lambda x} dx = \frac{K}{\lambda} e^{-\lambda s}$$

And notice that by this construction, $P(x \geq 30) = \frac{K}{\lambda} e^{-30\lambda} \approx 1$. We have designed our p -value estimation this way since strip scores below 30 are empirically observed to be unimportant.

Our next natural step, after obtaining p -values for each segment pair, is to provide a p -value estimate ζ for coincidental probability for the whole alignment, determined by the strips found. As mentioned earlier, we have learned that both r and t depend on sequence lengths m and n . Hence, if R and T are supposed to be the number of segment pairs and the total score of the segment pairs after adjusting for mean and variance based on sequence length, then the coincidental probability $\zeta = P(x \geq T, y \leq R)$. More specifically, ζ is the coincidental probability of seeing a total score of at least T using at most R segment pairs.

Figure 3 shows the distribution of r and t values observed from randomly generated sequences after adjusting for mean and variance. From it, we approximate for T and R that $P(x = T, y = R) = e^c e^{-a_t T^2 + b_t T + c_t} e^{-a_r R^2 + b_r R + c_r}$, where $c = -183.90$, $a_t = 10.1$, $b_t = 9070$, $c_t = -2.04 \times 10^6$, $a_r = 0.241$, $b_r = 4.71$, $c_r = -27.5$. This gives us for ζ that⁴:

$$\begin{aligned} \zeta &= P(x \geq T, y \leq R) = \\ &= \int_T^{\infty} \int_0^R e^c e^{-a_t x^2 + b_t x + c_t} e^{-a_r y^2 + b_r y + c_r} dy dx = \end{aligned}$$

⁴ Note that $\text{Erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-x^2} dx$

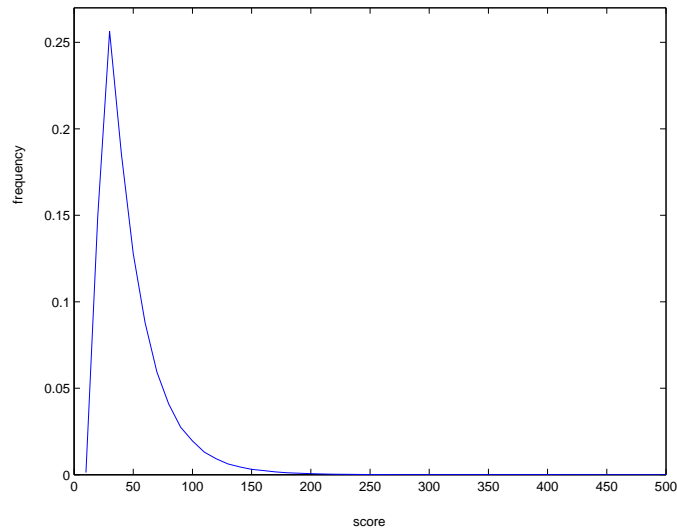


Fig. 2. Shown here is a plot of segment scores to frequency for randomly generated sequences using our assumption that segment score is length-independent. The x axis represents segment score, and the y axis represents frequency. The tail of this plot is an exponential distribution of form $P(S = x) = Ke^{-\lambda x}$, where we have approximated $K = 8.69 \times 10^{-2}$ and $\lambda = 3.26 \times 10^{-2}$. This curve is at its highest when $x = 30$, and by empirical observation, we have noticed that strips scoring less than 30 are generally unimportant portions of an alignment.

$$= \frac{\pi e^{c+c_t+c_r+\frac{b_t^2}{4a_t}+\frac{b_r^2}{4a_r}}}{4\sqrt{a_t a_r}} \left(1 - \text{Erf}\left(\frac{-b_t + 2a_t T}{2\sqrt{a_t}}\right)\right) \left(\text{Erf}\left(\frac{-b_r + 2a_r R}{2\sqrt{a_r}}\right) - \text{Erf}\left(\frac{-b_r}{2\sqrt{a_r}}\right)\right)$$

Furthermore, table 1 shows a comparison of alignments for biologically related sequences in terms of unadjusted r and t values, and ζ' values, all using $\rho = 0.5$. Note that $\zeta' = -\ln(\zeta)$. The conversion from ζ to ζ' was carried out for convenience in comparing lab results, where higher ζ' indicates results that are less coincidental. We chose to use $\rho = 0.5$ in all data shown in this table because with it, SEPA successfully filters away all segment pairs when aligning randomly generated DNA sequences, while retaining important segment pairs when aligning biologically related noncoding sequences, even when they have expected high gaps and low similarity regions. For further information regarding the sequences used, see Table 2 in the appendix.

Also, PLAINS does not always yield the results of least coincidental probability in this table, and this anomaly has a simple explanation. Note that the nature of PLAINS is to capture the biology faithfully even when the sequences have expected large gaps and low similarities. Thus it tries to aggressively align as many regions as possible, and hence in these situations, it produces r and t values that tend to be higher than those from other tools, even though its high r causes its overall result to appear more coincidental in spite of the compensating higher t . However, it turns out that when we fix r for all the tools, PLAINS yields higher t and hence better ζ' results. In other words, for any given r , each of the r segment pairs generated by PLAINS have smaller individual coincidental probabilities than the best r segment pairs generated by other tools. Figure 4 explains the details further.

3 Conclusions and Future Work

Our empirical analysis leads us to the conclusion that the SEPA-based p -value technique models coincidental probabilities much more accurately than the earlier technique employed in [3]. Furthermore, we note that aggressively incorporating too many segment pairs into an alignment can corrupt the overall result with false positives, in spite of an apparent improvement in the total score,

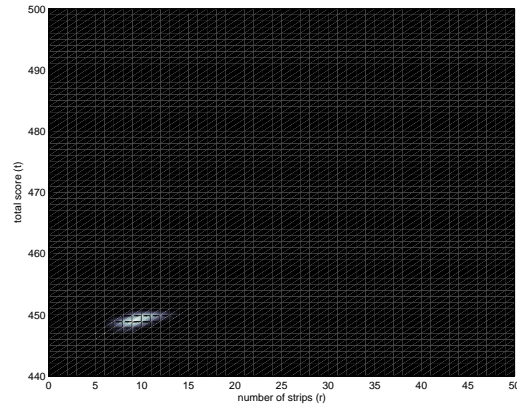


Fig. 3. From our alignments over the randomly generated sequences, after adjusting the number of segments r and the total score t for length-dependent average and deviation behavior, we chose to plot the frequency of observing certain r and t values. The figure shown here is a surface plot of this, where lighter spots indicate higher frequencies. From it, we observe that the majority of the data is concentrated in one area. This area approximates to $e^c e^{-a_t T^2 + b_t T + c_t} e^{-a_r R^2 + b_r R + c_r}$, where $c = -183.90$, $a_t = 10.1$, $b_t = 9070$, $c_t = -2.04 \times 10^6$, $a_r = 0.241$, $b_r = 4.71$, $c_r = -27.5$.

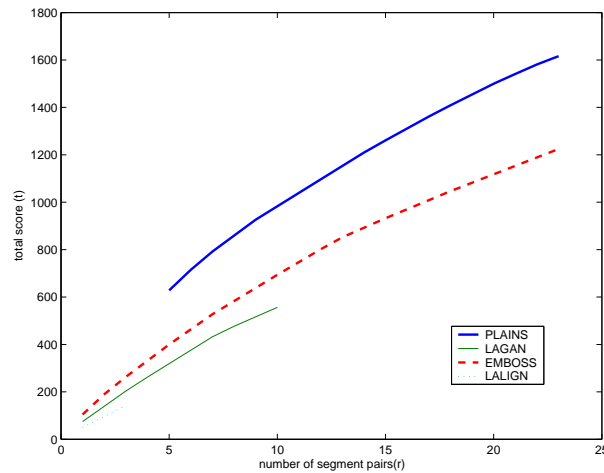


Fig. 4. In this figure, we observe the unadjusted r and t values produced by PLAINS, LAGAN, EMBOSS, and LALIGN from the human-mouse.3 – 9 experiment where we vary the ρ variable used to filter our segment pairs. On each curve, we observed the t and r values of each tool when varying ρ over various values from 0.1 till 0.9. Recall from table 1 that PLAINS performed poorly in terms of ζ' values for $\rho = 0.5$ for the human-mouse.3 – 9 experiments. However, note from this plot that for any fixed r where PLAINS is comparable to a different tool, PLAINS receives the highest t value, and therefore if we designed SEPA using a fixed r value over all alignment tools, then PLAINS would have the highest t value, and hence the highest ζ' value (i.e., the best result). Many other experiments from table 1 have a similar plot to this one.

Test Name	PLAINS			LAGAN			EMBOSS			LALIGN		
	t	r	ζ'	t	r	ζ'	t	r	ζ'	t	r	ζ'
HumanPseudo1	356.71	4	7.37	340.32	4	6.00	340.19	4	5.99	106.11	1	5.92
HumanPseudo2	285.75	3	3.96	281.84	3	3.94	238.30	3	3.87	105.02	1	6.14
HumanPseudo3	2181.50	14	47.18	441.58	6	22.98	1708.51	10	18.49	642.23	3	-0.00
HumanPseudo4	511.99	7	3.85	2172.40	14	-Inf	296.84	4	4.59	127.85	1	7.73
HumanPseudo5	792.64	7	7.29	775.74	7	7.29	176.73	1	13.04	184.59	1	13.04
MousePseudo1	389.84	4	13.97	386.88	4	13.40	388.88	4	13.78	174.45	1	5.86
MousePseudo2	461.68	6	8.88	453.64	6	7.77	206.02	2	5.56	208.76	2	5.56
MousePseudo3	72.19	1	6.75	72.19	1	6.75	83.34	1	6.75	84.22	1	6.75
fugu2r	534.14	5	11.15	360.22	3	13.05	151.39	2	14.07	186.37	2	14.07
HFortho1	734.82	7	10.94	349.33	4	14.18	374.35	5	13.05	x	x	x
HFortho2	600.22	4	16.78	555.61	4	16.78	327.91	1	20.18	307.82	2	19.01
HFortho3	637.52	7	14.53	259.44	3	19.05	409.99	5	16.71	x	x	x
HFortho4	1004.97	10	21.74	529.16	5	-0.00	367.86	4	-0.00	x	x	x
HFortho5	739.71	7	11.07	450.93	5	13.07	453.61	5	13.07	x	x	x
human_mouse.1_1	676.29	10	8.46	52.36	1	18.29	186.98	2	17.00	x	x	x
human_mouse.1_3	552.55	6	15.14	406.79	6	15.14	429.51	6	15.14	x	x	x
human_mouse.3_9	1260.69	15	15.47	432.25	7	24.23	801.15	12	18.44	x	x	x
human_mouse.3_16	218.47	3	5.71	x	x	x	180.05	2	6.77	64.33	1	7.93
human_mouse.4_3	262.19	3	15.44	74.91	1	17.79	176.83	2	16.59	x	x	x
human_mouse.4_5	421.71	6	7.35	221.57	3	10.47	401.71	5	8.32	x	x	x
human_mouse.6_17	986.89	12	23.00	240.10	3	-0.00	260.66	4	-0.00	x	x	x
human_mouse.7_11	594.32	8	9.06	164.10	2	15.44	476.71	7	9.99	x	x	x
human_mouse.17_11	608.75	7	13.93	171.96	3	18.57	451.60	6	15.02	x	x	x
human_mouse.x_x	1302.49	18	17.20	636.82	9	-0.00	568.46	9	-0.00	72.76	1	-0.00
human_dog.6_1	1239.35	14	18.99	424.59	6	-0.00	688.81	8	26.81	x	x	x
human_dog.6_12	1284.79	14	13.88	548.19	7	21.23	394.04	6	22.44	130.06	2	-0.00
human_dog.6_34	1488.26	16	-0.00	496.14	6	-0.00	900.73	12	-0.00	56.67	1	-0.00
human_dog.7_16	1042.19	13	10.45	128.07	2	22.40	309.03	4	19.84	x	x	x

Table 1. Shown here for PLAINS, EMBOSS, LAGAN, and LALIGN are the r , t , and ζ' values obtained from aligning genomic DNA sequences of lengths between 0.5 Kb and 12 Kb within human, mouse, dog, and fugu, where the pairs are biologically related and mainly noncoding DNA with expected large gaps and low homology regions. Please note the loss of precision involved in reporting ζ' values. Hence, if for a particular alignment, PLAINS and LAGAN receive ζ' values that differ by less than 1×10^2 , then their ζ' values would “appear” equal in this table.

as illustrated by PLAINS. However, SEPA can modify the overall alignment to select only the best r segments from an alignment while keeping the confidence in the final result high. It is here that the strength of PLAINS becomes obvious, since its r segments are less coincidental than its competition, and have higher scores, and hence better ζ' values.

However, in spite of the promising results from SEPA, there is still plenty of room for further improvements by using random portions of DNA from Human, Mouse, and Fugu instead of randomly generated DNA sequences. In that case, our concern shifts from the coincidental probability of a segment's score from aligning random DNA, to the coincidental probability of a segment's score from aligning unrelated random regions of organisms under comparison. Further extension includes development of better statistics that realistically capture the base-pair and coding/noncoding distributions within the sequences, as well as the effects of secondary and tertiary structures.

References

1. Altschul, S.F., Boguski, M.S., Gish, W., Wooton, J.C.: Issues in Searching Molecular Sequence Databases. *Nature Genetics* **6** (1994) 119–128
2. Brudno, M., Do, C., Cooper, G., Kim, M.F., Davydov, E., Green, E.D., Sidow, A., Batzoglou, S.: LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA. *Genome Research* **13**(4) (2003) 721–731
3. Gill, O., Zhou, Y., Mishra, B.: Aligning Sequences with Non-Affine Gap Penalty: PLAINS Algorithm, a Practical Implementation, and its Biological Applications in Comparative Genomics. *Series in Mathematical Biology and Medicine* **8** (2005). An unabridged version can be found at: <http://bioinformatics.nyu.edu/~gill/index.shtml>
4. Gu, X., Li, W.H.: The size distribution of insertions and deletions in human and rodent pseudogenes suggests the logarithmic gap penalty for sequence alignment. *J. Mol. Evol.* **40**(4) (1995) 464–473
5. Huang, X., Miller, W.: *Advanced Applied Mathematics* **12** (1991) 373–381
6. Iglehart, D.L.: Extreme Values in the GI/G/1 Queue. *The Annals of Mathematical Statistics* **43** (2) (1972) 627–635
7. Karlin, S., Altschul, S.F.: Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl. Acad. Sci. USA* **87** (1990) 2264–2268
8. Karlin, S., Altschul, S.F.: Applications and statistics for multiple high-scoring segments in molecular sequences. *Proc. Natl. Acad. Sci. USA* **90** (1993) 5873–5877
9. Karlin, S., Dembo, A., Kawabata, T.: Statistical Composition of High-Scoring Segments from Molecular Sequences. *The Annals of Statistics* **18** (2) (1990) 571–581
10. Ophir, R., Graur, D.: Patterns and rates of indel evolution in processed pseudogenes from humans and murids. *Gene*. **205**(1-2) (1997) 191–202
11. Pearson, W.R.: Comparison of Methods for Searching Protein Sequence Databases. *Protein Science* **4** (1995) 1145–1160
12. Pearson, W.R.: Searching Protein Sequence Libraries: Comparison of the Sensitivity and Selectivity of the Smith Waterman and FASTA algorithms. *Genomics* **11** (1991) 635–650
13. Rice, P., Longden, I., Bleasby, A.: EMBOSS: the European Molecular Biology Open Software Suite. *Trends Genetics* **Jun 16**(6) (2000) 276–277
14. Siegmund, D., Yakir, B.: Approximate p -Values for Local Sequence Alignments. *The Annals of Statistics* **28** (3) (2000) 657–680
15. Smith, T.F., Waterman, M.S.: Identification of Common Molecular Subsequences. *Journal of Molecular Biology* **147** (1981) 195–197
16. Shpaer, E., Robinson, M., Yee, D., Candlin, J., Mines, R., Hunkapiller, T.: Sensitivity and Selectivity in Protein Similarity Searches: A Comparison of Smith-Waterman in Hardware to BLAST and FASTA. *Genomics* **38** (1996) 179–191
17. States, D.J., Gish, W., Altschul, S.F.: Basic Local Alignment Search Tool. *Journal of Molecular Biology* **215** (1990) 403–410
18. Zhang, Z., Gerstein, M.: Patterns of nucleotide substitution, insertion and deletion in the human genome inferred from pseudogenes. *Nucleic Acids Res.* **31**(18) (2003) 5338–5348

Appendix

A Segment Pair Analysis in Further Detail

In order to approximate an appropriate p -value estimation for SEPA, we analyzed segment pairs behavior over our assumed null hypothesis of alignments for randomly generated nucleotide sequences. For length values ranging from 1000 bp to 8000 bp, we generated 25 random sequences. We also generated 25 random sequences of length 500 bp. For each combination of these length pairs, we ran all 625 possible pairwise alignments using PLAINS, and analyzed results using SEPA where $\rho = 1$ (to avoid filtering any segments out due to low p -value), and recorded the results in fig. 5, 6, 7, and 8.

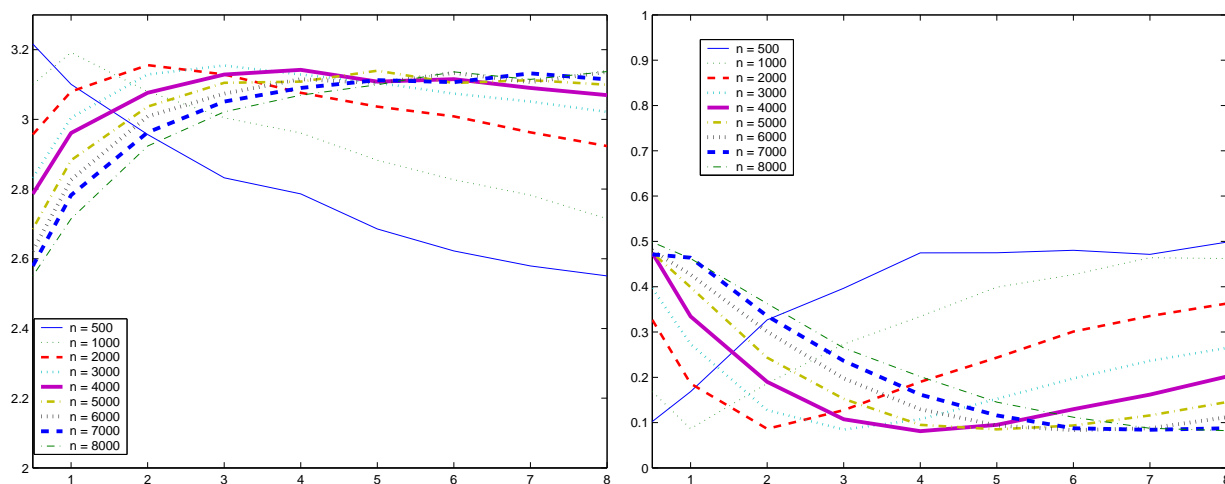


Fig. 5. Shown above are the mean and variance plots for the segment pair length-to-score ratio from aligning randomly generated DNA sequences. A unique line is plotted corresponding to each value of n in the thousand lengths ranging from 1000 to 8000. For these figures, and others that follow, x represents the m value divided by 1000, and y represents the mean or variance value obtained for that particular m and n . These plots indicate that, for the most part, the mean takes a constant value at 3.1, and the variance remains below 0.4, leading us to infer that length-to-score ratio can be well-approximated by a constant.

B Sequence Details

Shown in Table 2 are further details for the sequences used to compare PLAINS against LAGAN, EMBOSS, and LALIGN. Please note that sequences are expressed in their regular format unless they end with a “:-1” or “-” symbol, which indicates that they have been reverse-complemented prior to performing any alignments.

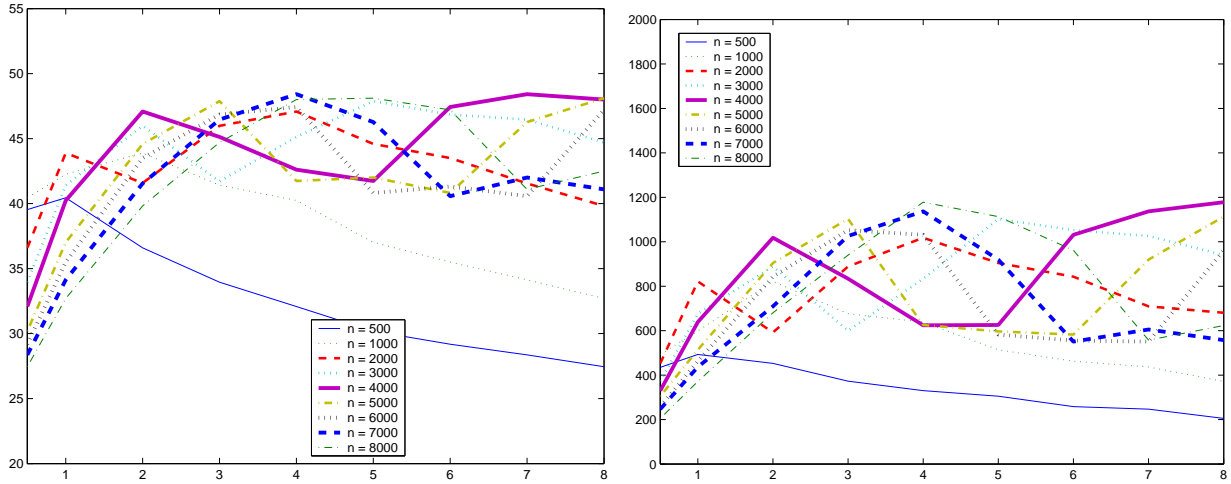


Fig. 6. Shown here are the mean and variance plots for segment scores from aligning randomly generated DNA sequences. From this we infer that, although for small n values, the average score decreases with increasing m , asymptotically (for large n) the average stays roughly constant in terms of m , while the variance fluctuates wildly around a constant value. Hence, for our scoring method, we model the segment scores as independent of m and n .

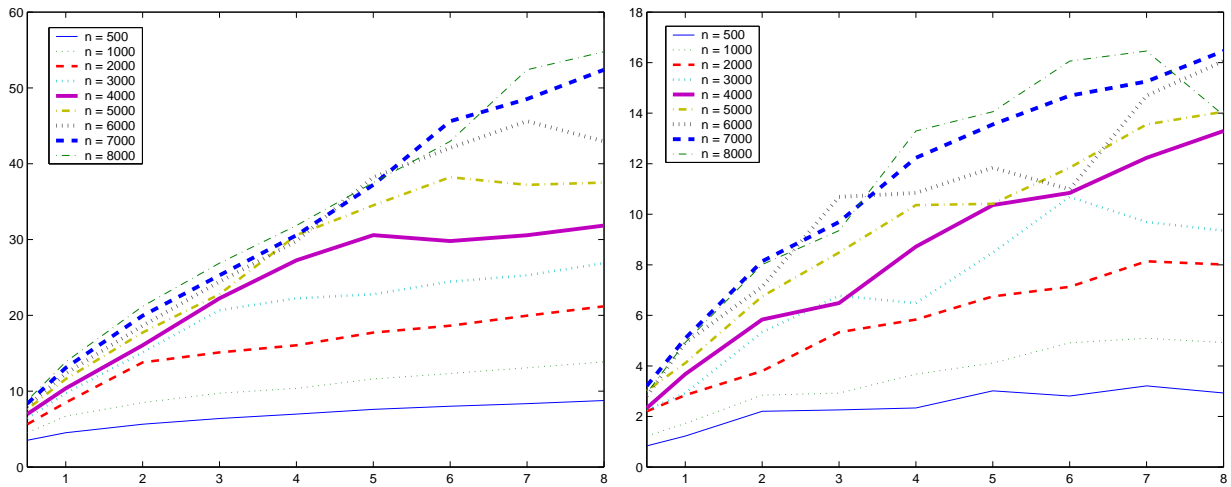


Fig. 7. Shown here are the mean and variance plots for r , the number of segment pairs obtained from aligning randomly generated DNA sequences. From this, we estimate that the average and variance of r scale roughly with $\Theta(\log(mn))$. More specifically, we approximate the mean of r and the variance of r , called $r_a(m, n)$ and $r_v(m, n)$ respectively, to scale roughly as $k_0 \ln(k_1 mn + k_2(m + n) + k_3)$ where k_0 , k_1 , k_2 , and k_3 are empirically determined constants. In the case of $r_a(m, n)$, we observe that $k_0 = 10^3$, $k_1 = 7.95 \times 10^{-10}$, $k_2 = 1.54 \times 10^{-7}$, $k_3 = 1.01$, and in the case of $r_v(m, n)$, we observe that $k_0 = 10^3$, $k_1 = 1.93 \times 10^{-10}$, $k_2 = 1.97 \times 10^{-7}$, $k_3 = 1.00$.

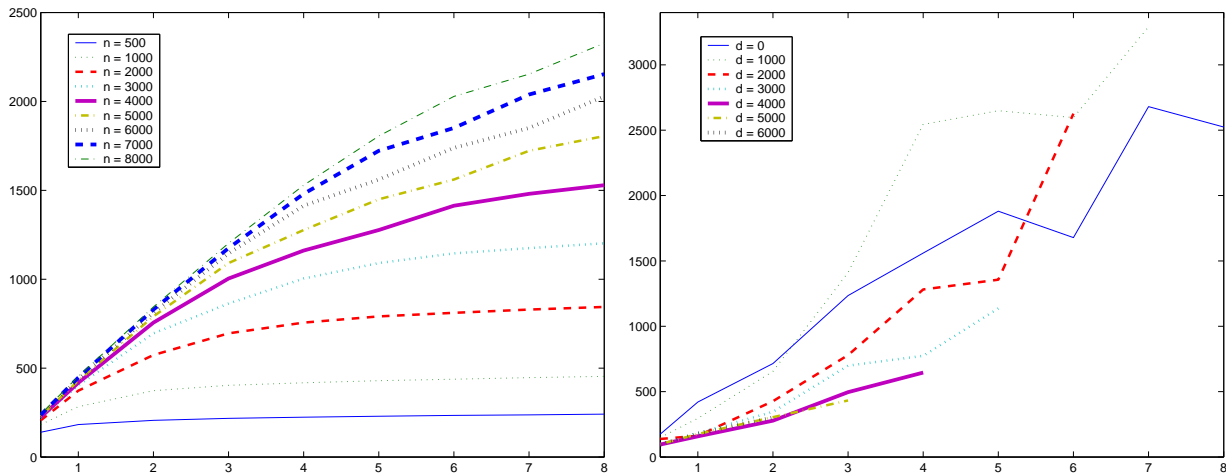


Fig. 8. The plots shown here are the mean and deviation plots for t , the total score of all segment pairs from aligning randomly generated DNA sequences. From this, we estimate that the average total score scales roughly with $\Theta(\log(mn))$. Because the variance plot was difficult to quantify in terms of m and n , we instead model the deviation for total score in terms of d and i , where $i = \min(m, n)$ and $d = \|m - n\|$. The lower figure shows the deviation plot, with each curve corresponding to a unique d value, and the x -axis representing i in units of thousands. From this, it was found that the deviation scales roughly with $\Theta(i \cdot d)$, but never declines below 100. More specifically, we approximate the average total score $t_a(m, n)$ to scale roughly as $k_0 \ln(k_1 mn + k_2(m + n) + k_3)$, and the deviation for total score $t_D(m, n)$ to scale roughly as $\max(k_0, k_1 i \cdot d + k_2 i + k_3 d + k_4)$, where k_0, k_1, k_2, k_3 , and k_4 are empirically estimated constants (and the variance $t_v(m, n) = t_D(m, n)^2$). In the case of $t_a(m, n)$, we observe that $k_0 = 10^5$, $k_1 = 4.29 \times 10^{-10}$, $k_2 = 1.33 \times 10^{-8}$, and $k_3 = 1.00$, and in the case of $t_v(m, n)$, we observe that $k_0 = 100$, $k_1 = -5.54 \times 10^{-5}$, $k_2 = 4.63 \times 10^{-1}$, $k_3 = 1.04 \times 10^{-2}$, and $k_4 = -65.01$.

Name	First Sequence	Second Sequence
HumanPseudo1	chr1 8257472 8257969 +	NCBI34:19:54160379:54161804:1
HumanPseudo2	chr1 163548408 163549002 +	NCBI34:4:174948678:174951482:-1
HumanPseudo3	chr1 212839737 212843396 +	NCBI34:19:47480657:47491789:1
HumanPseudo4	chr2 215849936 215850977 -	NCBI34:12:52960755:52965297:1
HumanPseudo5	chr3 154761512 154762855 -	NCBI34:20:62845714:62856853:-1
MousePseudo1	chr1 6930250 6930693 +	NCBIM32:4:116062392:116064688:1
MousePseudo2	chr10 34897773 34898331 +	NCBIM32:3:111151293:111157009:1
MousePseudo3	chr1 101195551 101195966 +	NCBIM32:19:41974653:41984383:1
fugu2r	NCBI34:6:10803176:10817954:1	FUGU2:scaffold_3266:7199:8502:1
HFortho1	NCBI34:22:17268346:17274146:1	FUGU2:scaffold_115:304567:308251:1
HFortho2	NCBI34:22:19452941:19466562:1	FUGU2:scaffold_385:130429:132429:1
HFortho3	NCBI34:21:31952480:31961633:1	FUGU2:scaffold_492:107025:110089:-1
HFortho4	NCBI34:4:78536922:78549607:1	FUGU2:scaffold_1018:38886:42563:-1
HFortho5	NCBI34:1:23574363:23584195:1	FUGU2:scaffold_2020:1332:3570:1
human_mouse.1_1	hg17 chr1:1045045-1049199	mm6 chr1:58087808-58093089 -
human_mouse.1_3	hg17 chr1:109911-115784	mm6 chr3:108302834-108307402 +
human_mouse.3_9	hg17 chr3:920975-927750	mm6 chr9:13034270-13040751 -
human_mouse.3_16	hg17 chr3:40927-45344	mm6 chr16:36425494-36426630 +
human_mouse.4_3	hg17 chr4:1016348-1026634	mm6 chr3:43806778-43808958 +
human_mouse.4_5	hg17 chr4:33206-37263	mm6 chr5:116454347-116457564 -
human_mouse.6_17	hg17 chr6:1515792-1522464	mm6 chr17:5319541-5327318 +
human_mouse.7_11	hg17 chr7:253979-256656	mm6 chr11:47406997-47414401 -
human_mouse.17_11	hg17 chr17:203511-209188	mm6 chr11:46304241-46308929 -
human_mouse.x_x	hg17 chrX:928373-936336	mm6 chrX:100457186-100463788 +
human_dog.6_1	hg17 chr6:48183-58637	canFam1 chr1:66683762-66688436 -
human_dog.6_12	hg17 chr6:791946-797744	canFam1 chr1:58385127-58391875 +
human_dog.6_34	hg17 chr6:1248975-1255904	canFam1 chr34:40546832-40556432 -
human_dog.7_16	hg17 chr7:40725-45009	canFam1 chr16:22868000-22875215 +

Table 2. Sequence Details for the Biologically Related Alignments Ran. All the sequences are retrieved from ENSEMBL database [www.ensembl.org].