Journal of Computational Biology 10(3-4):537-554

Significance Of Inter-Species Matches When Evolutionary Rate Varies

Jia Li * Webb Miller †

Abstract

We develop techniques to estimate the statistical significance of gap-free alignments between two genomic DNA sequences, using human-mouse alignments as an example. The sequences are assumed to be sufficiently similar that some but not all of the neutrally evolving regions (i.e., those under no evolutionary constraint) can be reliably aligned. Our goal is to model the situation in which the neutral rate of evolution, and hence the extent of the aligning intervals, varies across the genome. In some cases, this permits the weaker of two matches to be judged as less likely to have arisen by chance, provided it lies in a genomic interval with a high level of background divergence. We employ a Hidden Markov Model to capture variations in divergence rates, and assign probability values to gap-free alignments using techniques of Dembo and Karlin, which are related to those used for the same purpose by Blast. Our methods are illustrated in detail using a 1.49 Mb genomic region. Results obtained from the analysis of human chromosome 22 using these techniques are also provided.

Key words: DNA sequence alignment, p-values of inter-species matches, evolutionary rate Hidden Markov model, Human Chromosome 22

I Introduction

Aligning human and mouse genomic sequences has been proposed as a high-throughput strategy for analyzing and annotating the human genome. In particular, a genomic interval that is highly conserved between the two species can be considered as a candidate for encoding a protein [14] or regulating gene transcription [12]. The proposal has been adopted whole-heartedly by the genomics community, resulting in accelerated programs to sequence the mouse genome by Celera Genomics and, independently, by the public sequencing consortium. Mouse whole-genome shotgun sequence data in the public domain are just beginning to be used for improving the analysis and annotation of the public sequence data from the human genome.

There exists no uniquely plausible criterion for determining whether a genomic interval is "highly conserved". Of course, part of the difficulty lies in the fact that any threshold will be at least somewhat arbitrary. A more vexing problem stems from regional differences in the

*Statistics Department, Penn State, University Park, PA 16802; jiali@stat.psu.edu; Phone: 814-863-3074; Fax: 814-863-7114

†Department of Computer Science and Engineering, Penn State, University Park, PA 16802; webb@bio.cse.psu.edu

background level of human-mouse similarity. Human-mouse evolutionary separation occurred only about 80 million years ago, which is so recent that many freely evolving regions (i.e., under no apparent evolutionary constraint) can be reliably aligned across at least some of their length. However, the fraction of apparently unconstrained DNA that can be aligned is highly dependent on genomic location [10, 8], with one study [8] finding that the percentage of non-repetitive (e.g., not Alu or L1 sequences) and non-coding DNA that can be aligned varied from 11% (in the ERCC2 region) to 99% (in the HOXA cluster).

A number of authors have observed variability of divergence rates. Wolfe *et al.* [22] found that the rate of silent substitutions in protein-coding regions (i.e., nucleotide changes that do not affect the encoded amino acid sequence) varies widely among genes. These authors and others have concluded that silent substitutions are neutral, or nearly so, implying that the rate of neutral evolution is highly variable, depending on position in the genome. Koop [17] observed that some comparisons of non-coding DNA between humans and mice show a very high level of conservation in presumably non-functional intervals, other regions show very low conservation, and still others are intermediate. Matassi *et al.* [19] show that silent substitution rates in two genes separated by at most one centiMorgan (roughly one or two megabases) are more similar to each other on average than are a randomly chosen gene pair, suggesting that genomic domains of similar neutral evolutionary rate may exist on a megabase scale.

This variability makes it difficult to give an objective and appropriate criterion for deciding if a genomic interval deserves to be classified as "more conserved than can be expected by chance alone". For instance, consider the three panels in Figure 1 from a "percent identity plot", or PIP. This PIP provides a graphical summary of a few of the local alignments of 1.49 Mb of DNA sequence from the velocardiofacial syndrome (VCFS) region of human chromosome 22, for which almost all of the orthologous mouse sequence is available [18]. Within those local alignments, three gap-free segments of roughly comparable lengths and percent identities are highlighted. Based just on these lengths and identities, it is difficult to rank their relative strengths, particularly if one wants to account for the very different degrees of apparent background divergence among the rows of Figure 1. The point of this paper is to provide an objective and statistically rigorous method for ranking the matches according to which of them is less likely to have arisen by chance.

It is critical that this variation in rate of neutral evolution be better quantified and understood. Genomic intervals identified as "highly conserved" are candidates for a number of experimental tests for functionality, including tests to see whether they are expressed as genes or regulate such expression. Such experiments, particularly those for regulatory function, are expensive and tedious, so it is important that identification of candidate regions have as rigorous a basis as possible. It is particularly desirable that this be done with a strong statistical underpinning, e.g., to quantify the extent to which an interval is more conserved than can be explained by chance.

Our approach to aligning genomic sequences begins by computing a set of local alignments between sequences of genomic DNA, with the intention of capturing precisely the detectable homologies. For the VCFS data, we began by identifying interspersed repeats in the human sequence using the RepeatMasker program (A. Smit and P. Green, unpublished), then removing from the human sequence all interspersed repeats that we believe to have inserted after the human-mouse split. In addition, all annotated exons were removed, since our primary interest is in finding functional non-coding intervals. Removing these two classes of segments reduced the 1.49 Mb sequence to about 1.06 Mb. Older repeats were "soft masked", i.e., not allowed

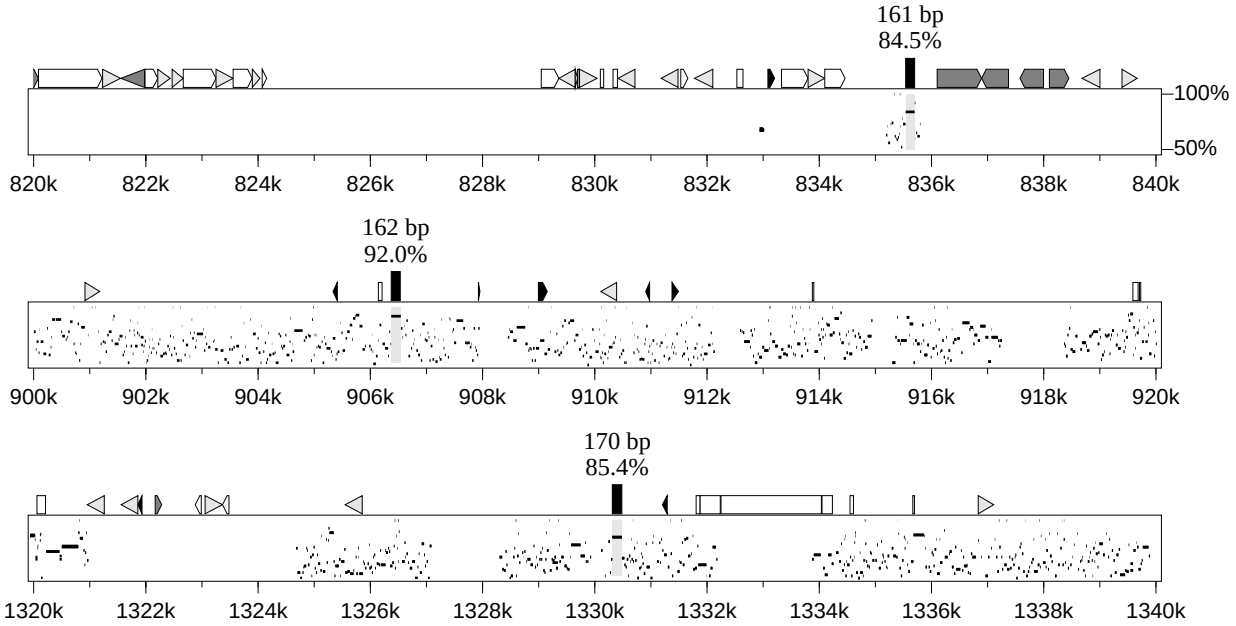


Figure 1: Percent identity plot (PIP) of some human-mouse alignments. Triangles and other icons along the top indicate positions of interspersed repeats and low-complexity regions found in the human sequence by the RepeatMasker program. Each tiny horizontal line in the PIP indicates the human positions and percent nucleotide identity of an interval between consecutive gaps in a local alignment with the mouse genomic sequence.

to align in preliminary computations that determine the rough locations of alignments, but allowed to align during the final (gapped alignment) phase. Alignments were computed by the *blastz* program [20] with default parameters.

Our statistical analysis of these alignments is performed in two phases. First, we use a hidden Markov model (HMM) to detect long-range patterns in the regional variation of divergence level. The basic goal is to identify a few classes of genomic regions according to the extent that the human sequence can be reliably aligned with mouse data, and to do so in a statistically sound manner that makes a minimal number of *a priori* assumptions. The second phase describes the alignments in each of these classes with a Markov model (as distinct from a hidden Markov model), which is used to determine statistical significance in a manner appropriate for the level of divergence seen in that type of genomic region. Significance is expressed as a p-value, giving the probability that a match of equal or higher score could happen by chance. We now sketch these two phases in turn.

With our approach to aligning genomic sequences, regional variation in divergence level is revealed most directly by differences in the percentage of nucleotides covered by local alignments, rather than by the percent of nucleotide identity within alignments. For instance, in one study [10], human-mouse alignments showed a spread of 6.4% to 78.1% in the fraction of non-repetitive, non-coding DNA that aligns, but with only a spread of 64.3% to 75.0% in percentage of nucleotide identity within those local alignments. Indeed, the two regions with the highest percent identity had the lowest fraction of aligning DNA, indicating that percent identity is a poor discriminator of divergence level. Accordingly, to embody divergence level, we temporarily set aside information about the internal structure of local alignments and rep-

resented the genomic region by a 1.11 Mb sequence of 0's and 1's; it alternates between runs of 0's, with a 0 for every unaligned human position, and runs of 1's, with a 1 for every column of a local alignment. (Thus, a local alignment generates a number of 1's that exceeds the length of the aligned human segment by the total length of inserted mouse nucleotides, i.e., gaps in the human sequence.)

Our approach to training an HMM, as described in the next section, models the sequence of 0's and 1's with four "states", which can be thought of as "modes" hidden in the data. Table 1 summarizes some of the states' characteristics. When in one of the states, the sequence consists almost entirely of 0's (i.e., unaligned), in the second it is almost entirely 1's, and the other two have intermediate frequencies (23.8% and 72.1%) of 0's. The last row of Table 1 gives the state's stationary frequency.

Assigning p-values to gap-free alignments, such as the three intervals highlighted in Figure 1, requires modeling the internal structure of alignments. To do this, we replace each 0 in the 1.11 Mb sequence by `U` (for Unaligned), and replace each 1 by either `G` (for position in a Gap), `M` (for Matched to an identical symbol) or `N` (Non-match). Given a long gap-free alignment, we want to determine the probability that an equivalent or stronger run of consecutive Match and Non-match positions could happen by chance in a region with a similar degree of alignability. This provides a "p-value" for any strong gap-free alignment, analogous to those made popular in bioinformatics by Blast [1]. Indeed, the theoretical underpinnings [16] of our approach to p-values are a generalization to Markov models of the Karlin-Altschul method, as originally described [15] for the simpler case of a sequence generated from independent identical distributions (i.i.d).

State	1	2	3	4
unaligned	99.1%	23.8%	72.1%	0.14%
occupied	32.0%	26.8%	35.4%	5.7%

Table 1: Characteristics of the HMM's four states. First row: the percentage of unaligned base pairs in each state. Second row: the stationary frequency of each state.

Table 2 presents the p-values, as computed by the methods described here, of the three segments highlighted in Figure 1. Compared to the third segment, the first segment is somewhat shorter and has a marginally lower percentage of matched base pairs, hence its score is lower, e.g., 111 vs. 121 if we score 1 for a match and –1 for a non-match. However, because the first segment is located in a region with poorer alignability, it is statistically more significant than the third segment, as indicated by the p-values.

Segment	1	2	3
Length	161	162	170
Percentage of `M`	84.5%	92.0%	85.4%
score (`M`=1, `N`=–1)	111	137	121
p-value	0.0075	0.0013	0.0081

Table 2: The p-values of the three segments indicated in Figure 1.

The next two sections cover the details of the two phases of our statistical analysis, i.e., training an HMM to model the high-level variation in divergence and computing p-values, respectively. Readers of those sections are assumed to be familiar with basic concepts of proba-

bilistic analysis of DNA sequences, roughly at the level of the first three chapters of the book by Durbin, Eddy, Krogh and Mitchison [9]. The most difficult details are placed in the Appendix. At the end, we discuss generalizations of our results that remain to be explored and suggests ways that our methods may be used to obtain insight into several basic questions concerning the mechanisms and tempo of evolutionary processes.

II Modeling with an HMM

As described in the previous section, we represent a set of local alignments of a "reference" nucleotide sequence with some other sequence as a sequence of symbols `U`, `G`, `M` and `N`, whose length exceeds that of the reference sequence by the total lengths of all gaps in that sequence within a local alignment. For the first phase of the statistical analysis, we ignore the distinction between `G`, `M` and `N`, and work with a sequence of 0's (instead of `U`) and 1's (instead of `G`, `M` and `N`). The goal of this section is to extract "modes" from this sequence of 0's and 1's.

A classic HMM for this sequence, in which each state emits either 0 or 1 according to a certain distribution determined solely by the state, would work poorly. The problem lies in the fact that the run lengths of 0's and 1's are very large. The histograms of the run lengths of 0's and 1's are provided in Figure 2. Since states in an HMM represent modes of context in the sequence, we expect a region of a fixed state to cover multiple runs of 1's and 0's. Otherwise, the separation of the sequence into regions of different states is over-localized. For a region in state m, every observation 0 or 1 is generated independently according to the probability mass function $(p_0(m), p_1(m))$. The expected run lengths of 0's and 1's are $1/p_1(m)$ and $1/p_0(m)$ respectively. Hence, if $p_1(m)$ and $p_0(m)$ are not very close to the extreme values 0.0 and 1.0, the expected run lengths of both 1 and 0 cannot be very large. For example, if $p_1(m) < 0.9$, the expected run length of 1 is no greater than 10, and the probability of a run length being larger than 50 is only 0.005. However, as indicated in Figure 2, a vast majority of the run lengths in the sequence are longer than 50. For the run lengths of 1's, 99.98% of them are longer than 50. We thus expect an HMM that fits the sequence well tends to have states with either very high values or very low values of $p_1(m)$. The long runs of 1's are generated by states with $p_1(m)$ close to 1.0 and the long runs of 0's are generated by states with $p_1(m)$ close to 0.0. The estimated underlying states switch from one to another almost in synchronization with the switch from a run of 1's to a run of 0's or vice versa. Such an HMM thus provides little information regarding the context in the sequence.

The inappropriateness of the basic HMM to model the alignment sequence is also demonstrated by experiments. For instance, we trained a basic HMM with 11 states on the entire sequence of human chromosome 22. Except for one state with probability of occurring in the sequence as low as 6.4×10^{-9}, all the others fall into two groups. Those in the first group have $p_1(m) > 0.999$ and those in the second have $p_1(m) < 1.0 \times 10^{-15}$. This HMM extracts essentially two modes of context: nearly zero percent of alignment or nearly perfect alignment. The strong dependence among adjacent positions forces the HMM to be over-localized since the Markovian property assumed about states is the only mechanism to account for the inter-position dependence.

We propose an extended HMM that takes into consideration the strong inter-position dependence and in the mean time is capable of extracting modes of context. In the new model, the underlying states are assumed to be first order Markovian, just as in the basic HMM. The difference lies in the conditional distribution of observations given states. In the basic HMM,

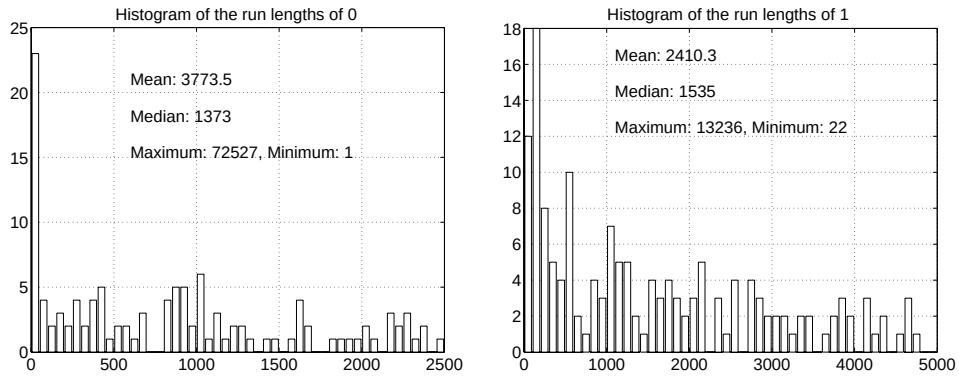


Figure 2: Histograms of the run lengths of 0's and 1's in the alignment sequence. The ranges of the histograms displayed are not complete.

it is assumed that given all the states, the conditional distribution of observation x_t at position t only dependents on the state s_t at the same position, that is,

$$P(x_1, x_2, ..., x_T \mid s_1, s_2, ..., s_T) \quad = \quad P(x_1 \mid s_1)P(x_2 \mid s_2) \cdots P(x_T \mid s_T) \; . \tag{1}$$

In the new model, we assume that given all the states, the conditional distribution of observation x_t at position t depends on the observation x_{t-1} and state s_{t-1} at the previous position. In a sense, the conditional independence assumption on observations in the basic HMM is replaced by a conditional first order Markovian assumption. This Markovian property of observations embedded in a state allows long runs to occur without compromising the role of a state in representing context. Equation (1) is changed to

$$P(x_1, x_2, ..., x_T \mid s_1, s_2, ..., s_T) \quad = \quad P(x_1 \mid s_1)P(x_2 \mid x_1, s_1) \cdots P(x_T \mid x_{T-1}, s_{T-1}) \; .$$

For an HMM with M states, we need to estimate $2M$ probability mass functions: $P(x_t = i \mid x_{t-1} = j, s_{t-1} = m)$, $i, j = 0, 1$, $m = 1$,...,M, as well as the state transition probability matrix, $||a_{m,n}||$, $m, n = 1, ..., M$. For notational simplicity, we write $p_{j,i}(m) = P(x_t = i \mid x_{t-1} = j, s_{t-1} = m)$. The new HMM is referred to as the HMM with Markovian observations (HMMMO).

The structure of the model allows us to use a modified version of the Baum-Welch algorithm [3]. Computation time is proportional to the product of the sequence length and the number of states; so is the memory consumed. When dealing with sequences of length in the order of tens of millions, a few gigabytes of memory are required. However, the memory requirement can be reduced to an order proportional to the square root of the sequence length if we double the amount of computation. We present here the original version of the estimation algorithm.

To estimate an HMM with Markovian observations by the maximum likelihood criterion, the EM algorithm [2, 3] is applied. The EM algorithm estimates a model by updating the parameters iteratively. Let $L_m(t)$ denote the conditional probability of being in state m at position t given all the observations, and $H_{m,n}(t)$ denote the conditional probability of a transition from state m at position t to state n at position $t + 1$ given all the observations, both computed from a current set of parameters estimated. The re-estimation formulae for the transition probabilities

$a_{m,n}$, m,$n = 1,...,M$, and the probabilities $p_{j,i}(m)$, i,$j = 0,1$, $m = 1,...,M$, are

$$\begin{aligned} p_{j,i}(m) &= \frac{\sum_{t=1}^{T-1} L_m(t) I(x_t = j) I(x_{t+1} = i)}{\sum_{t=1}^{T-1} L_m(t) I(x_t = j)} \\ a_{m,n} &= \frac{\sum_{t=1}^{T-1} H_{m,n}(t)}{\sum_{t=1}^{T-1} L_m(t)} , \end{aligned}$$

where as usual $I(\cdot)$ is the indicator function that equals 1 when the argument is true and zero otherwise. The probabilities $L_m(t)$ and $H_{m,n}(t)$ can be computed efficiently by the *forward-backward* algorithm. This algorithm was developed as a part of the Baum-Welch algorithm. For the HMMMO, because of the Markovian assumption on observations given states, the forward-backward algorithm is modified correspondingly.

Define the forward probability $\alpha_m(t)$ as the joint probability of observing the first t x_τ's, $\tau = 1, ..., t$, and being in state m at position t. This probability can be evaluated by the following recursive formula

$$\begin{aligned} \alpha_m(1) &= \pi_m p_{x_1}(m) \quad 1 \le m \le M \\ \alpha_m(t) &= \sum_{n=1}^{M} \alpha_n(t-1) p_{x_{t-1},x_t}(n) a_{n,m} \\ &\quad 1 < t \le T,\ 1 \le m \le M . \end{aligned}$$

The probabilities π_m, $m = 1,...,M$ are the initial probabilities of the M states, which can be derived from the transition probability matrix if we assume π_m as the stationary frequency of state m. The probabilities $p_i(m)$, $i = 0,1$, $m = 1,...,M$ are the marginal probabilities of the observations 0 and 1 in state m. Assuming $p_i(m)$ as the stationary frequency of i in state m, we can compute it from the conditional distributions $p_{j,i}(m)$, $j = 0,1$.

Define the backward probability $\beta_m(t)$ as the conditional probability of observing x_τ's after position t, $\tau = t+1, ..., T$, given the state at position t is m and the observation at t is x_t. As with the forward probability, the backward probability can be evaluated using the following recursion

$$\begin{aligned} \beta_m(T) &= 1 \\ \beta_m(t) &= p_{x_t,x_{t+1}}(m) \sum_{n=1}^{M} a_{m,n} \beta_n(t+1) , \quad 1 \le t < T . \end{aligned}$$

The probabilities $L_m(t)$ and $H_{m,n}(t)$ are solved by

$$\begin{aligned} L_m(t) &= P(s_t = m \mid \mathbf{x}) = \frac{P(\mathbf{x}, s_t = m)}{P(\mathbf{x})} = \frac{1}{P(\mathbf{x})} \alpha_m(t) \beta_m(t) \\ H_{m,n}(t) &= P(s_t = m, s_{t+1} = n \mid \mathbf{x}) = \frac{1}{P(\mathbf{x})} \alpha_m(t) a_{m,n} p_{x_t,x_{t+1}}(m) \beta_n(t+1) , \end{aligned}$$

where $P(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m(\tau)\beta_m(\tau)$ is the joint probability of observing all x_t's, $t = 1, ..., T$. The summation is identical for all τ's. Hence, in particular, $P(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m(T)$.

The number of states in the HMM is chosen by the Bayesian Information Criterion (BIC) [21]. By BIC, the optimal model maximizes the penalized log likelihood $\log P(\mathbf{x}) + \frac{k}{2} \log T$, where k,

increasing with the number of states, is the number of parameters in the HMM. For an HMMMO with M states, the number of parameters to specify the transition matrix $||a_{m,n}||$ is $M(M-1)$; and the number of parameters needed to describe the Markov chain of 0 and 1 in each state is 2. Hence the total number of parameters in an HMMMO is $M(M-1)+2M = M^2+M$. The number of states selected by BIC for the VCFS sequence is 4. Constraints can also be put on the state transition probabilities $a_{m,n}$ to reduce the complexity of an HMM. For instance, we may constrain $a_{m,n}$ to be the same for all $n \neq m$. One motivation to use the constrained model is that the estimated values of $a_{m,n}$, $n \neq m$, are often at the order of 10^{-5} or smaller. A sequence of length around one million, such as the VCFS region, cannot provide sufficient amount of data to estimate these $a_{m,n}$. More robust estimation can be achieved by using the model with reduced complexity. The preference to the constrained model is also supported by BIC as the penalized log likelihood of this model is higher than that of the unconstrained one.

According to the HMM, a subsequence $\{x_{t_1}, x_{t_1+1}, ..., x_{t_2}\}$ with states fixed as m is a Markov chain switching between two symbols: 0 and 1. The transition probabilities of the Markov chain are $p_{i,j}(m)$, $i,j = 0,1$. Matched, Non-matched and Gap base pairs are not distinguished by this Markov chain. To model the alignment sequence with 4 possible symbols, we assume that within a run of 1's in state m, the sequence of `M`, `N`, and `G`, denoting Matched, Non-matched and Gap correspondingly, is a Markov chain. The transition probabilities of the Markov chain vary with state m. The initial probabilities of `M`, `N`, and `G` are $\pi_{\mathtt{M}}(m)$, $\pi_{\mathtt{N}}(m)$, and $\pi_{\mathtt{G}}(m)$. We have $\pi_{\mathtt{G}}(m) = 0.0$ since at the initial position of a run of 1's, Gap never occurs. Denote the transition probabilities between `M`, `N`, and `G` within a run of 1's in state m by $\bar{a}_{\gamma,\zeta}(m)$, $\gamma,\zeta = \mathtt{M},\mathtt{N},\mathtt{G}$, $m = 1,2,...,M$. Denote the sequence of 4 symbols by $\{y_1, y_2, ...y_T\}$. The sequence $\{x_1, x_2, ..., x_T\}$ is formed by setting $x_t = 0$ if $y_t = \mathtt{U}$ (Unaligned), and $x_t = 1$ otherwise. Given the HMM trained from $\{x_t\}_{t=1}^T$, the maximum likelihood estimation of $\bar{a}_{\gamma,\zeta}(m)$ is

$$\bar{a}_{\gamma,\zeta}(m) = \frac{\sum_{t=1}^{T-1} L_m(t) I(y_t = \gamma) I(y_{t+1} = \zeta)}{\sum_{\zeta} \sum_{t=1}^{T-1} L_m(t) I(y_t = \gamma) I(y_{t+1} = \zeta)}$$

$$\pi_{\gamma}(m) = \frac{\sum_{t=1}^{T-1} L_m(t) I(x_t = 0, x_{t+1} = 1) I(y_{t+1} = \gamma)}{\sum_{t=1}^{T-1} L_m(t) I(x_t = 0, x_{t+1} = 1)} \qquad \gamma,\zeta = \mathtt{M},\mathtt{N},\mathtt{G}.$$

The above estimation is essentially the computation of the empirical frequencies of all the transitions. Each count is weighted by the posterior probability of being in state m at the corresponding position if the Markov chain to be estimated is for state m.

To sum up, within one state m, the sequence of 4 symbols is modeled by two embedded Markov chains. The first Markov chain specifies statistically the transition between 0 (unaligned base pairs) and 1 (all the other possible base pairs). The second Markov chain specifies the transition between `M`, `N`, and `G` within a run of 1's. It is straightforward to see that the statistical model characterized by these two embedded Markov chains is equivalent to a Markov chain with the 4 symbols: `U`, `M`, `N`, and `G`. Denote the transition probabilities of this Markov chain by $a_{\gamma,\zeta}(m)$, $\gamma,\zeta = \mathtt{U},\mathtt{M},\mathtt{N},\mathtt{G}$. Then,

$$a_{\gamma,\zeta}(m) = \begin{cases} p_{0,0}(m) & \gamma = \zeta = \mathtt{U} \\ p_{0,1}(m)\pi_{\zeta} & \gamma = \mathtt{U},\ \zeta \neq \mathtt{U} \\ p_{1,0}(m) & \gamma \neq \mathtt{U},\ \zeta = \mathtt{U} \\ p_{1,1}(m)\bar{a}_{\gamma,\zeta}(m) & \gamma \neq \mathtt{U},\ \zeta \neq \mathtt{U} \end{cases}.$$

Denote the Markov chain in state m by $\mathcal{P}_m$.

III Significance of High-Scoring Segments

Given a long gap-free alignment composed of only matched and non-matched base pairs, we are concerned with its statistical significance. We assign each possible symbol $\gamma \in \{\mathtt{U}, \mathtt{M}, \mathtt{N}, \mathtt{G}\}$ a score Z, e.g., $Z_{\mathtt{M}} = 1$, $Z_{\mathtt{N}} = -1$, $Z_{\mathtt{U}} = Z_{\mathtt{G}} = -L$, where $L \gg 1$. The score of a segment of symbols is defined as the sum of the scores of all the positions in the segment. A long gap-free alignment with high percentage of $\mathtt{M}$ yields a high segment score.

Theorems in Karlin and Dembo [16] laid the foundation for assessing the statistical significance of a high-scoring segment in a Markov chain. These theorems enable us to compute the limit probability of a sequence generated randomly by the Markov chain having its maximal segment score exceeding that of a given segment, or in some cases, to obtain bounds for the probability. The limit is taken with the sequence length approaching infinity. A low probability value indicates the high score of the segment is statistically significant. Being imprecise with terminology, we refer to the probability as the p-value of the segment for simplicity. If we want to test the null hypothesis that a sequence is generated by a Markov process, the maximal segment score of the sequence will be identified. The null hypothesis is rejected with p-value equal to the probability of a sequence generated randomly by the same Markov process containing segments with higher scores. Next, we discuss the application of theorems in [16] to our scoring scheme.

Consider a Markov chain $\mathcal{P}$ with r possible symbols, denoted by $\{Y_1, Y_2, ..., Y_T\}$, $Y_t \in \Gamma = \{\zeta_1, \zeta_2, ..., \zeta_r\}$. Let the transition probability matrix of the Markov chain be $\mathbf{P} = ||p_{\gamma\zeta}||$, $\gamma, \zeta \in \Gamma$. For each transition from symbol ζ_i to ζ_j, a score Z_{ζ_i,ζ_j} is assigned. Assigning a score to each symbol ζ_j can be viewed as a special case of assigning scores to transitions, in which Z_{ζ_i,ζ_j} are the same for a fixed ζ_j and different ζ_i's. Given a realization $\{y_t\}_{t=1}^T$ of the Markov chain with $y_0 = \gamma$, the score of a segment $\{y_{t_1}, ..., y_{t_2}\}$ is $\sum_{\tau=t_1}^{t_2-1} Z_{y_\tau, y_{\tau+1}}$; and the maximal segment score is $M_\gamma(T) = \max_{1 \le t_1 \le t_2 \le T} \sum_{\tau=t_1}^{t_2-1} Z_{y_\tau, y_{\tau+1}}$.

To apply theorems in [16], we make the following assumptions about the Markov chain $\mathcal{P}$:

1. The Markov chain $\mathcal{P}$ is irreducible and aperiodic.

2. The negative drift condition is

$$E[Z] = \sum_{\zeta_i, \zeta_j} \pi_{\zeta_i} p_{\zeta_i,\zeta_j} Z_{\zeta_i,\zeta_j} < 0 \ ,$$

   where π_{ζ_i} is the stationary frequency of ζ_i, determined by the transition probability matrix $\mathbf{P}$. Note p_{ζ_i,ζ_j} is the transition probability specified in $\mathbf{P}$.

3. For each symbol ζ_i, there exists a symbol ζ_j such that $p_{\zeta_i,\zeta_j} > 0$, $Z_{\zeta_i,\zeta_j} > 0$; and a symbol ζ_k such that $p_{\zeta_i,\zeta_k} > 0$, $Z_{\zeta_i,\zeta_k} < 0$. Or more generally, there exists ζ_i and a sequence $y_0 = \zeta_i$, y_1, ..., $y_m = \zeta_i$, such that $P\{\sum_{\tau=0}^{k-1} Z_{y_\tau, y_{\tau+1}} > 0, k = 1, ..., m-1 \mid y_0 = y_m = \zeta_i\} > 0$

If scores Z_{ζ_i,ζ_j}, $i = 1, 2, ..., r$, are non-lattice [16], then

$$\lim_{T \to \infty} P\{M_\gamma(T) - \frac{\ln T}{\theta^*} > z\} = 1 - \exp(-K^* e^{-\theta^* z}) \ ,$$

where K^* and θ^* are constants determined by the transition probability matrix $\mathbf{P}$ and the scores Z_{ζ_i,ζ_j}. Define matrix

$$\Phi(\theta) = ||p_{\gamma,\zeta} e^{\theta Z_{\gamma,\zeta}}|| \ .$$

The constant θ^* is the unique positive solution of the equation $\rho(\theta) = 1$, where $\rho(\theta)$ is the dominant eigenvalue of the matrix $\Phi(\theta)$. Algorithms for computing constants K^* and θ^* are presented in Appendix.

If scores Z_{ζ_i,ζ_j} are lattice of span δ (δ is the largest number of which all the Z_{ζ_i,ζ_j}'s are multiples),

$$\begin{aligned} & 1 - \exp(-K^* e^{-\theta^* z}) \\ \leq & \liminf_{T\to\infty} P\{M_\gamma(T) - \frac{\ln T}{\theta^*} > z\} \\ \leq & \limsup_{T\to\infty} P\{M_\gamma(T) - \frac{\ln T}{\theta^*} > z\} \\ \leq & 1 - \exp(-K^+ e^{-\theta^* z}) \end{aligned} \quad (2)$$

where $K^+ = e^{\theta^*\delta} K^*$.

For integer scores Z_{ζ_i,ζ_j} with the maximum common divisor equal 1, the scores are lattice of span $\delta = 1$. We use Inequality (2) to compute the p-value of a high-scoring segment. When T is sufficiently large, $P\{M_\gamma(T) - \frac{\ln T}{\theta^*} > z\}$ is bounded between $1 - \exp(-K^* e^{-\theta^* z})$ and $1 - \exp(-K^+ e^{-\theta^* z})$. For a high-scoring segment with score $\frac{\ln T}{\theta^*} + z$, the upper bound $1 - \exp(-K^+ e^{-\theta^* z})$ provides a "conservative" p-value for the segment. Since $K^+ = e^{\theta^*} K^*$, K^+ is close to K^* when θ^* is close to zero. In this case, the upper bound is close to the real probability $P\{M_\gamma(T) - \frac{\ln T}{\theta^*} > z\}$.

For the alignment sequence of 4 symbols $\{y_t\}_{t=1}^T$, we focus on the special case of assigning a score to each symbol. For brevity, we denote the scores by $Z_{\mathtt{M}} = 1$, $Z_{\mathtt{N}} = -1$, $Z_{\mathtt{U}} = Z_{\mathtt{G}} = -L$, $L \gg 1$. Based on the HMM trained from the sequence of 0's and 1's, i.e., $\{x_t\}_{t=1}^T$, a Markov chain characterizing $\{y_t\}_{t=1}^T$ is estimated within each state of the HMM. For the 4-state HMM, optimal according to BIC, the Markov chain within each state is irreducible and aperiodic. The negative drift condition is satisfied with sufficiently large L. We set $L = 400$ in our experiment. When L is very large, a high-scoring segment cannot contain any `G` or `U`, since one such symbol can lower the score of the entire segment to a negative value. Therefore, a segment with a high score is simply a long gap-free alignment with a high percentage of `M`. Therefore, the exact score values assigned to `G` and `U` have little effect on p-values, which is demonstrated by experiments.

The constants K^*, K^+, and θ^* of Markov chains $\mathcal{P}_m$, $m = 1, ..., 4$ are listed in Table 3. The percentage of unaligned base pairs in each state, indicating alignability, and the stationary frequency of each state are presented in Table 1.

State	1	2	3	4
θ^*	0.0959	0.1142	0.1132	0.1079
K^*	0.0001	0.0261	0.0085	0.0325
K^+	0.0001	0.0292	0.0095	0.0362

Table 3: Constants of the Markov chain in each state. These constants are used for computing p-values.

Assume a state sequence $\{s_t\}_{t=1}^T$ is generated randomly according to the Markov chain governing the states of an HMM. The initial probabilities of the states are the stationary frequencies of the states π_m, $m = 1, 2, ..., M$, where $M = 4$ for the HMM trained from the VCFS sequence. A sequence $\{y'_t\}_{t=1}^T$ of symbols `U`,`G`,`M`, and `N` are generated based on the state

sequence $\{s_t\}_{t=1}^{T}$. Within a region of a fixed state m, $t_1 \le t \le t_2$, the sequence $\{y'_t\}_{t=t_1}^{t_2}$ is generated by the corresponding Markov chain in state m, $\mathcal{P}_m$. When $T \to \infty$, with probability 1, the percentage of positions in state m is π_m. Hence when T is large, the number of positions in state m is approximately $\pi_m T$. By Inequality (2), the probability that the maximal segment score of positions in state m exceeds $z + \frac{\ln \pi_m T}{\theta^*(m)}$ is bounded as follows

$$1 - \exp(-K^*(m)e^{-\theta^*(m)z}) \le P\{M_\gamma(T,m) > z + \frac{\ln \pi_m T}{\theta^*(m)}\} \le 1 - \exp(-K^+(m)e^{-\theta^*(m)z})$$

Notation $\theta^*(m)$, $K^*(m)$, and $K^+(m)$ are used to stress that these constants are determined by the Markov chain $\mathcal{P}_m$. The fact that positions in state m may not be consecutive is ignored as the average run length of one state is much larger than the lengths of high-scoring segments we consider. In the sequel, sequences in discussion are assumed to be realizations of the 4-state HMM with a Markov chain embedded in each state.

The p-value as a function of the segment score $\bar{z}$ for the Markov chain of each state m is

$$p_v(\bar{z}, m) = 1 - \exp(-K^+(m)e^{-\theta^*(m)(\bar{z} - \frac{\ln \pi_m T}{\theta^*(m)})}) \ ,$$

which is plotted in Figure 3.

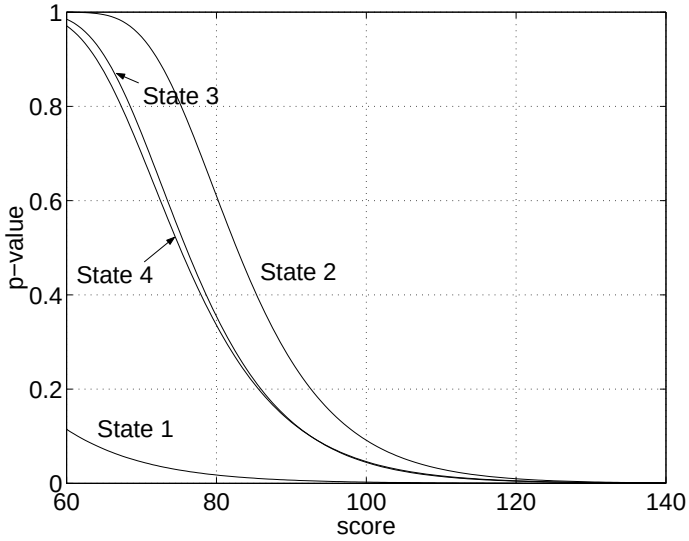


Figure 3: The p-value as a function of the segment score for the Markov chain in each state.

Suppose we compare the statistical significance of two high-scoring segments in two regions with great difference in divergence rate. To take into account the "background" difference of the two segments, for each region, we compute the conditional probability distribution of the state of a randomly selected position from the region given the entire observed sequence. Assume a region ranges from t_1 to t_2 and t is a position randomly selected from the region, the conditional probability $P\{s_t = m \mid y_1, y_2, ..., y_T\}$ is

$$P\{s_t = m \mid y_1, y_2, ..., y_T\} = \frac{\sum_{\tau=t_1}^{t_2} L_m(\tau)}{t_2 - t_1 + 1} \ ,$$

where $L_m(\tau)$ is the conditional probability of position τ being in state m given the observed sequence $\{y_1, y_2, ..., y_T\}$. The forward-backward algorithm is used to compute $L_m(\tau)$. If a state is selected randomly according to the distribution $P\{s_t = m \mid y_1, y_2, ..., y_T\}$, $m = 1, 2, ..., M$, then the probability of the maximal segment score of positions in the state exceeding a score $\bar{z}$ is the weighted sum of the p-values $p_v(\bar{z}, m)$, that is,

$$\bar{p}_v(\bar{z}) = \sum_{m=1}^{M} P\{s_t = m \mid y_1, y_2, ..., y_T\} p_v(\bar{z}, m) \ .$$

We use $\bar{p}_v(\bar{z})$ as a measure of the statistical significance of a segment with score $\bar{z}$ in the region $t_1 \leq t \leq t_2$. For simplicity, we refer to $\bar{p}_v(\bar{z})$ as the p-value of a segment with score $\bar{z}$. Although $p_v(\bar{z}, m)$ decreases with the increase of $\bar{z}$ for all m, $\bar{p}_v(\bar{z})$, incorporating information about background alignability, may yield a lower value for smaller $\bar{z}$ if the segment locates in a region with poorer alignability and hence is more significant relative to its background.

To compute the p-value of a segment without its background region specified, we use a window centered around the segment as the background. The window size is the average run length of a state in the HMM. For the 4-state HMM trained from the VCFS sequence, the average number of positions staying in one state is about 14,200. Table 2 presents the p-values of three segments, which locate separately in the three PIP panels in Figure 1. Background regions used for the three segments are windows centered around them. Note that due to the poor alignability of its background region, the first segment has a smaller p-value than the third one although its score is lower.

IV Example: γ-globin regulatory element

This section illustrates the generality of our use of Markov models of alignments to assign a statistical significance to highly conserved regions. We show that the approach can be applied to multiple (as well as pairwise) alignments, and can be used to find regions with high levels of divergence (as well as similarity). We also illustrate some of the flexibility in assigning scores to columns.

In higher primates, the γ-globin gene is expressed in the fetus, whereas in lower primates it is expressed earlier during development. For instance, in humans (which have two nearly identical copies of the gene), it is expressed fetally, while lemurs express it in the embryo. To find the signals in genomic DNA responsible for this difference, one might align, say, the 1000 bp immediately upstream (a typical location for regulatory elements) of the α-globin gene for several higher primates and several lower primates, then look for regions where the sequences from the higher primates agree with each other but not with the lower primates.

We did just that, using the sequences from human, rhesus monkey and woolly monkey (higher primates), as well as tarsier and galago (lower primates). Sequences were aligned using the MultiPipMaker Web site [20]:

http://bio.cse.psu.edu/

Frequently, gene regulation is performed by transcription factors that permit a certain level of deviation from the consensus binding pattern. Hence, we did not want to require absolute identity among the three higher primate sequences. Define an alignment column to be of class 1 if the three higher primates have the same nucleotide and each lower primate has a different letter. A column is in class 2 if the three higher primates have two different nucleotides among

them and each lower primate has a different letter. All other columns are in class 3. We decided to give columns a score of 4, 2 or –1, if they are in class 1, 2 or 3, respectively.

We used an algorithm [13] that locates regions of high total score within the alignment. The algorithm runs in time proportional to the length of the alignment and identifies regions whose total score cannot be improved either by expanding or shrinking the run of columns. (In case of a tie, the algorithm picks the longer run.) The three highest scoring regions are as follows.

	region 1	region 2	region 3
human	AAAATTGGTACAT	GCTAAAGGGAAGAATAAATT	GGCGGCTGGCTAGGGATG
rhesus	.....	.....	.....
woolly	.......TC....	.......AAG.T.....	..T..G.....
tarsier	GTT...T..CT.G	A.C.....A.-----....G	AAA.---.T.A.AT..CA
galago	----------	----------	A.G.---...C.A....A
class	1113331331131	13133333331211133331	132312133313133331

Here we use a dot to indicate a nucleotide that is identical to the first one in its column. Dashes indicate a gap.

The regions have respective scores 22, 18 and 18, and approximate p-values 0.15, 0.39 and 0.39. The third region is known to be critical for the difference in expression patterns of the α-globin gene between higher and lower primates, and it was initially located by visually inspecting short pairwise alignments for sequence differences, a procedure that was called *differential phylogenetic footprinting* [11].

Other ways of scoring the alignment columns do a better job of emphasizing the third region. For instance, we can down-weight the potential contribution of gaps in the sequences of lower primates as follows. Consider a column formerly in class 1 or 2 to be in class 4 if contains a gap symbol that immediately follows a gap symbol in the same row. Give columns in class 4 the score 0, keeping other scores the same. Now, region 3 scores 12, which ties it with another region for the highest scoring segment within the aligning kilobase sequences, with an estimated significance of 0.48. This example illustrates the subtle issues that may be involved in selecting column scores that best reveal a desired biological phenomenon, particularly with multiple alignments.

V Human Chromosome 22

The above approach to computing p-values is used to analyze the entire sequence of human chromosome 22, consisting of roughly 47.7 M base pairs. The aligned sequence comprises five symbols: `U` (unaligned position), `G` (gap), `M` (match), `S` (transition), and `V` (transversion). The symbols `S` and `V` are both treated as `N` in the VCFS sequence. Scores assigned to these five symbols are $Z_{\mathtt{M}} = 1$, $Z_{\mathtt{S}} = -1$, $Z_{\mathtt{V}} = Z_{\mathtt{G}} = Z_{\mathtt{U}} = -3$.

An HMM is trained on the 0/1 sequence converted from the aligned sequence by setting `U` to 0 and all the other symbols to 1. The optimal number of states selected by BIC for the HMM is 4. Characteristics of the four states are summarized in Table 4, the first row being percentages of unaligned positions in the states and the second being the stationary frequencies of the states. The constants K^*, K^+, and θ^* for computing p-values within State 2, 3, and 4 are listed in Table 5. For State 1, the Markov chain is not irreducible because once it enters `U`, it will never transit to another symbol. Hence, the Markov chain in this state cannot yield a positive scoring segment when it becomes stationary. State 1 apparently results from the

nearly 13 M straight U's at the beginning of the chromosome 22 sequence. This state does not affect the computation of the p-values since its posterior probability given a region containing aligned segments is zero. If a three-state HMM is trained on the sequence with the beginning 13 M straight U's deleted, the three states are expected to be roughly the same as State 2, 3, and 4 in this four-state HMM trained from the entire sequence.

State	1	2	3	4
unaligned	99.94%	63.85%	86.84%	28.33%
occupied	28.89%	31.22%	19.73%	20.16%

Table 4: Characteristics of the four states in the HMM trained on the chromosome 22 sequence. First row: the percentage of unaligned base pairs in each state. Second row: the stationary frequency of each state.

State	2	3	4
θ^*	0.1276	0.1298	0.1115
K^*	0.0089	0.0028	0.0158
K^+	0.0101	0.0032	0.0177

Table 5: Constants of the Markov chains in 3 states of the HMM trained on the chromosome 22 sequence. These constants are used for computing p-values.

The p-value as a function of the score based on each Markov chain in State 2, 3, and 4 is plotted in Figure 4. Comparing with the other two states, State 4 corresponds to the most conserved background, so a segment with a fixed score arisen in this state is least significant, reflected by the highest p-value. State 3 has the highest divergence rate among the three and yields the lowest p-values consequently.

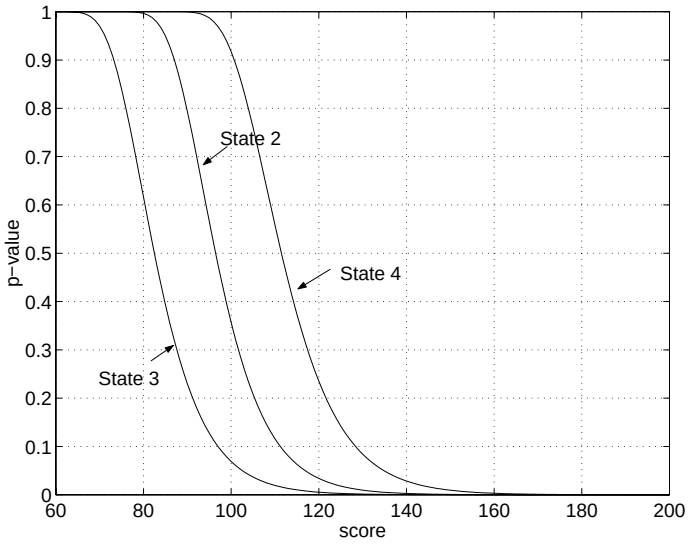


Figure 4: The p-value as a function of the segment score for the Markov chains in State 2, 3, and 4 of the chromosome 22 sequence.

To demonstrate the effect of different divergence rates on the computation of p-values, the p-values of a collection of segments scoring from 110 to 170 are plotted in Figure 5. This collection is not the complete set of segments in the chromosome 22 sequence with scores in that range. In the figure, for any given score, only segments with either very low p-values or very high p-values are shown. It is demonstrated that the p-value of a segment may differ enormously due to different levels of background conservativeness.

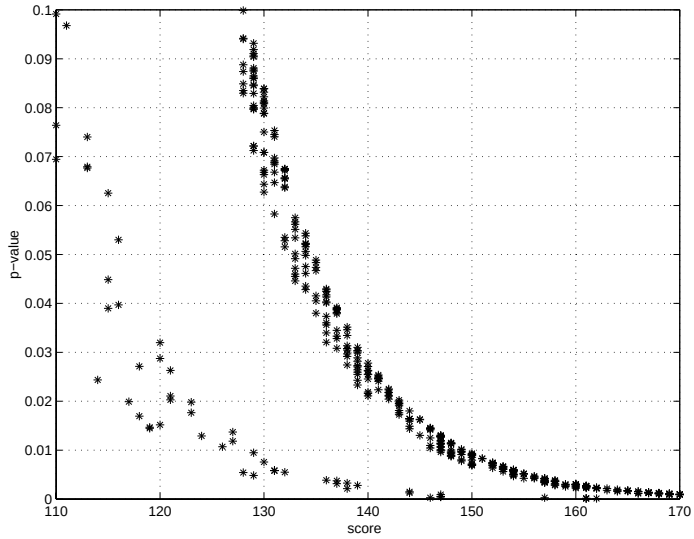
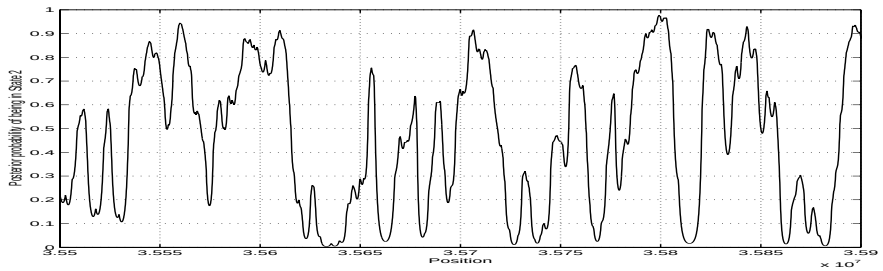


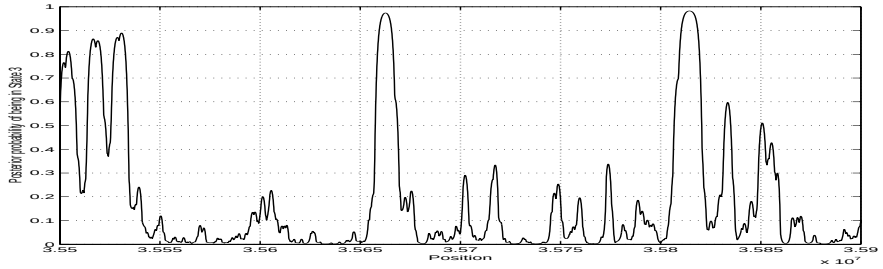
Figure 5: The p-values of a set of high scoring segments.

In Figure 6, an example region with a number of high scoring segments is shown. The first three panels plot the posterior probability of being in State 2, 3, and 4 at each base pair position given the entire sequence. State 1 is not displayed as the posterior probability of being in it is nearly zero across this entire region. In addition, the posterior probabilities of being in the four states sum up to 1 and hence possess only 3 degrees of freedom. To save memory, the posterior probabilities are averaged across every 100 base pairs. A large probability in State 4 indicates a highly conserved region. Segments with the same score tend to have higher p-values (less significant) if the posterior probability of being in State 4 is large. The fourth panel shows the scores of 8 segments of lengths around 200, marked at their center positions. The fifth panel shows the p-values of the 8 segments, the first three of which locate in a less conserved background than do the other five. The p-value of the third segment with score 144 is the lowest although three other segments have higher scores of 156, 153, and 161.

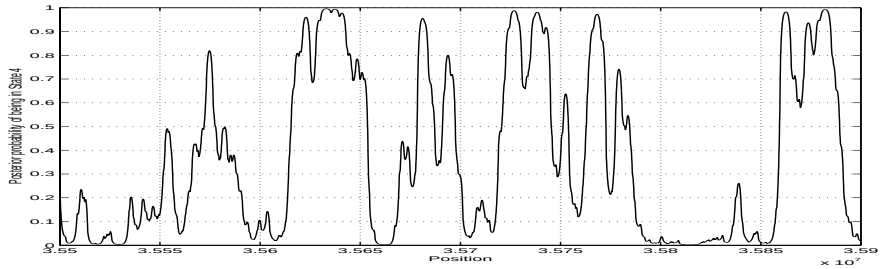
It takes roughly 6 hours to train the four state HMM for the chromosome 22 sequence on a 700MHz PC with Linux OS and about 7.1 minutes to train the embedded Markov chains within each state. To obtain p-values for segments in the aligned sequence, the posterior probability of being in each state at each base pair position needs to be computed. These posterior probabilities can be evaluated in one run, which takes about 7.25 minutes on the 700MHz PC, and stored for later use. The amount of time necessary to compute the constants θ^*, K^*, K^+ and to compute the p-value of a segment using those constants and the pre-stored posterior probabilities is negligible, substantially lower than 1 second.



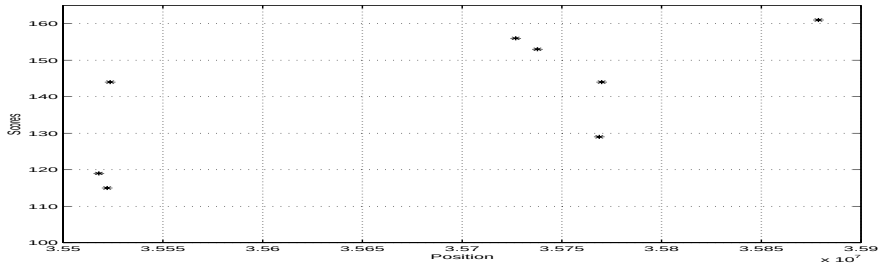
(a)



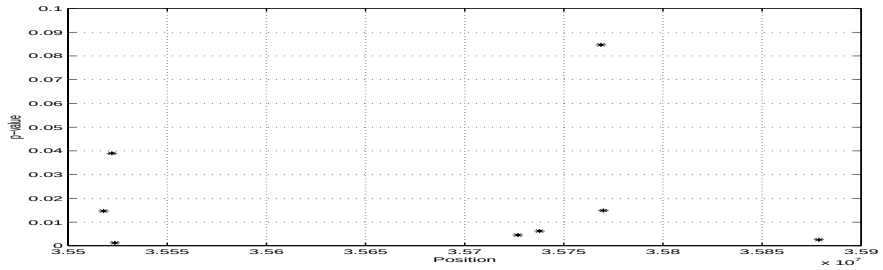
(b)



(c)



(d)



(e)

Figure 6: An example region in the aligned chromosome 22 from position 35.5M to 35.9M. (a)-(c): The posterior probability of being in State 2, 3, and 4 correspondingly, (d) The scores of high scoring segments displayed at their center positions in this region, (e) Corresponding p-values of these segments.

VI Discussion

Partitioning the genome along the lines discussed here has the potential of determining whether the human genome falls naturally into some number of divergence levels. Koop [17] detected three levels of human-mouse divergence in non-coding regions, but that observation was based on inspection of only five genomic loci. It remains to be seen whether the classification of genomic regions according to "alignability" is discrete or continuous. A natural comparison is with *isochores*, i.e., genomic regions of more-or-less constant percentage of `C` and `G` nucleotides (as opposed to `A` and `T`). The human genome has been asserted [4, 5] to fall into five isochore types, with isochores generally being at least 200 kb in length, and with fairly sharp boundaries between successive isochores. Even now, with the human genome sequence largely in hand, the theory remains controversial.

The existence of differences in evolutionary rate between different parts of the genome should not come as a complete surprise. It has been known for years that there are positional differences in the rates that DNA is damaged and repaired, a fact of considerable interest to those studying evolution [7] and cancer [6]. However, a mechanism that creates divergence-rate differences on a genome-wide scale has yet to be identified.

A natural way to begin seeking biological explanations for these differences is to ask whether divergence rate is correlated with other varying genomic properties, such as GC level, recombination rate, gene density, and position in the nucleus. An exciting prospect is that segmenting the human genome according to rate of sequence conservation with the mouse will reveal a pattern that provides a clue to the biological mechanism for the rate variation.

Acknowledgement

Ross Hardison, David Haussler and Amir Dembo provided helpful suggestions. Arian Smit determined how best to identify interspersed repeats and other elements that inserted after human-mouse divergence. Laura Elnitski assembled and annotated the VCFS sequences.

Appendix

We present here the algorithms for computing K^* and θ^* of a Markov chain with scores $Z_{\mathtt{M}} = n_1$, $Z_{\mathtt{N}} = -n_2$, $Z_{\mathtt{U}} = Z_{\mathtt{G}} = -m$, where $m, n_1 > 0$; $n_2 \geq 0$; and n_1 and n_2 are prime to each other, or $n_1 = 1$ and $n_2 = 0$. To constrain high-scoring segments to gap-free alignments, we require $m \gg n_1$ and $m \gg n_2$. For very large m, as discussed in Section III, the exact value of m has little effect on the constants, and hence p-values. We restrict the greatest common divisor of n_1 and n_2 to be 1 so that the span of scores is 1. If the greatest common divisor of n_1 and n_2 is not 1, we can always scale the scores by the common divisor. Any segment score is then scaled by the same factor, so the p-values can be computed with the new set of scores.

Readers are referred to Karlin and Dembo [16] for the general algorithms on computing K^* and θ^* with integer scores of span 1. With significant computational simplification, the algorithm for computing K^* provided here is an approximation to the general algorithm in [16]. Errors resulted from the approximation decay exponentially fast with m. For the scores we consider, m is usually in the order of hundreds. Consequently, imprecision caused by the approximation is negligible.

Denote the transition probability matrix of the Markov chain by $\mathbf{P} = ||p_{\gamma\zeta}||$. The element $p_{\gamma\zeta}$ in $\mathbf{P}$ is the probability of entering state ζ given that the current state is γ. The four symbols are put in the order `U`, `G`, `M`, `N`. For instance, the entry on the second row and the third column

of $\mathbf{P}$ is the transition probability from G to M. The score of symbol γ is Z_γ, $\gamma \in \mathcal{S} = \{\mathtt{U}, \mathtt{G}, \mathtt{M}, \mathtt{N}\}$. Define matrix $\Phi(\theta) = ||p_{\gamma\zeta} e^{\theta Z_\zeta}||$, that is, $\Phi(\theta)$ is obtained from $\mathbf{P}$ by multiplying its ζth column by $e^{\theta Z_\zeta}$. The stationary mean score $E[Z] = \sum_\gamma \pi_\gamma Z_\gamma$, where π_γ is the stationary frequency of symbol γ according to the transition probability matrix $\mathbf{P}$. $E[Z]$ is negative by assumption. Define $\mathcal{I}$ as the set of integers between $-m$ and n_1. Partition $\mathbf{P}$ in the form $\mathbf{P} = \sum_{i\in\mathcal{I}} \mathbf{P}^{(i)}$, where $\mathbf{P}^{(i)} = ||p^{(i)}_{\gamma\zeta}||$, $p^{(i)}_{\gamma\zeta} = I(Z_\zeta = i)p_{\gamma\zeta}$, where as usual $I(\cdot)$ is the indicator function. In particular $\mathbf{P}^{(i)} = \mathbf{0}$ if $i \neq n_1, -n_2, -m$. If we write $\mathbf{P} = (\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4)$, where $\mathbf{p}_j$'s are column vectors, then $\mathbf{P}^{(n_1)} = (\mathbf{0}, \mathbf{0}, \mathbf{p}_3, \mathbf{0})$, $\mathbf{P}^{(-n_2)} = (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{p_4})$, $\mathbf{P}^{(-m)} = (\mathbf{p}_1, \mathbf{p}_2, \mathbf{0}, \mathbf{0})$.

The algorithms for computing K^* and θ^* are outlined below. The third step, which is the key step for computing K^*, is expanded next.

1. Determine $\theta^* > 0$ such that $\rho(\theta^*) = 1$, where $\rho(\theta)$ is the maximum eigenvalue of matrix $\Phi(\theta)$. As $\rho(\theta)$ is convex [16], θ^* can by searched by a simple doubling and halving routine that converges rapidly.

2. Determine the right frequency eigenvector $\mathbf{u}^* = \mathbf{u}(\theta^*)$ of $\Phi(\theta^*)$.

3. Compute matrices $\mathbf{Q}^{(i)}$, $i = -1, -2, ..., -m$ and $\mathbf{Q} = \sum_{i=-m}^{-1} \mathbf{Q}^{(i)}$. Also compute $\mathbf{G}^{(j)}$, $j = 1, 2, ..., n_1$ and $\mathbf{G} = \sum_{j=1}^{n_1} \mathbf{G}^{(j)}$. $\mathbf{Q}^{(i)}$ and $\mathbf{G}^{(j)}$ are substochastic matrices; and $\mathbf{Q}$ and $\mathbf{G}$ are stochastic matrices. The computation of $\mathbf{Q}^{(i)}$ and $\mathbf{G}^{(j)}$ will be presented in a moment.

4. Determine the stationary frequency vectors of $\mathbf{Q}$ and $\mathbf{G}$, i.e., $\mathbf{zQ} = \mathbf{z}$ and $\mathbf{wG} = \mathbf{w}$.

5. Compute $K^* = v(\infty)c(\infty)$, where

$$c(\infty) = \frac{\langle \mathbf{w}, (\mathbf{I} - \sum_{j=1}^{n_1} \mathbf{G}^{(j)} e^{-\theta^* j}) \frac{1}{\mathbf{u}^*} \rangle}{\langle \mathbf{w}, (\sum_{j=1}^{n_1} j\mathbf{G}^{(j)})\mathbf{e} \rangle (e^{\theta^*} - 1)}$$

$$v(\infty) = \frac{\langle \mathbf{z}, (\mathbf{I} - \sum_{i=-m}^{-1} \mathbf{Q}^{(i)} e^{\theta^* i})\mathbf{u}^* \rangle E[Z]}{\langle \mathbf{z}, (\sum_{i=-m}^{-1} i\mathbf{Q}^{(i)})\mathbf{e} \rangle},$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{e} = (1, 1, 1, 1)^t$, and $\frac{1}{\mathbf{u}^*}$ denotes the vector formed by taking reciprocal of each element of $\mathbf{u}^*$, i.e., $(1/u_1^*, 1/u_2^*, 1/u_3^*, 1/u_4^*)^t$. The notation $\langle \cdot, \cdot \rangle$ means the inner product of the two vectors.

To compute $\mathbf{Q}^{(i)}$ and $\mathbf{G}^{(j)}$, we introduce the following definitions. Let $\hat{\mathbf{P}}^{(i)} = (\mathbf{I} - \mathbf{P}^{(0)})^{-1}\mathbf{P}^{(i)}$. If $n_2 \neq 0$, $\mathbf{P}^{(0)} = \mathbf{0}$. Hence $\hat{\mathbf{P}}^{(i)} = \mathbf{P}^{(i)}$. Let $\mathbf{D}_{\mathbf{u}^*} = diag(u_1^*, u_2^*, u_3^*, u_4^*)$ be the diagonal matrix with the components of $\mathbf{u}(\theta^*)$ on the diagonal, and $\hat{\mathbf{T}}^{(i)} = e^{\theta^* i}\mathbf{D}_{\mathbf{u}^*}^{-1}\hat{\mathbf{P}}^{(i)}\mathbf{D}_{\mathbf{u}^*}$. Note $\hat{\mathbf{T}}^{(i)} = \mathbf{0}$ if $i \neq n_1, -n_2, -m$.

Compute $\mathbf{G}^{(j)}$, $j = 1, 2, ..., n_1$ by the following recursive formula. $\mathbf{G}^{(l)} = \mathbf{0}$ for $l > n_1$.

1. $\mathbf{G}^{(j)}_{(1)} = \hat{\mathbf{T}}^{(j)}$.

2. $\mathbf{G}^{(j)}_{(k)} = \hat{\mathbf{T}}^{(j)} + \hat{\mathbf{T}}^{(-n_2)} \sum_{\mathbf{l}} \mathbf{G}^{(l_1)}_{(k-1)} \cdots \mathbf{G}^{(l_\sigma)}_{(k-1)}$. The sum is over the index range $1 \leq l_1, l_2, ..., l_\sigma \leq n_1$, $l_1 + l_2 + \cdots l_\sigma = j + n_2$ and $l_\sigma \geq j$, where σ is any positive integer that yields a valid set of $l_1, \cdots, l_\sigma$.

$\mathbf{G}_{(k)}^{(j)}$ converges to $\mathbf{G}^{(j)}$ geometrically. The recursive formula is an approximation to that in [16]. If $n_1 = 1$, $\mathbf{G}^{(1)}$ can be computed by the exact recursive formula

$$\mathbf{G}_{(k)}^{(1)} \;=\; \hat{\mathbf{T}}^{(1)} + \hat{\mathbf{T}}^{(-n_2)}(\mathbf{G}_{(k-1)}^{(1)})^{n_2+1} + \hat{\mathbf{T}}^{(-m)}(\mathbf{G}_{(k-1)}^{(1)})^{m+1} \; .$$

The third term in the sum is omitted in the approximation since $m \gg n_1$ and $m \gg n_2$.

Compute $\mathbf{G} = \sum_{j=1}^{n_1} \mathbf{G}^{(j)}$. For scores we consider, since $\hat{\mathbf{T}}^{(j)} = \mathbf{0}$, if $j \neq n_1$ and $j > 0$, and $\hat{\mathbf{T}}^{(n_1)}$ is of form $(\mathbf{0}, \mathbf{0}, \mathbf{p}, \mathbf{0})$, i.e., only the third column vector is nonzero, $\mathbf{G}^{(j)}$, $j = 1, 2, ..., n_1$, are also of form $(\mathbf{0}, \mathbf{0}, \mathbf{p}, \mathbf{0})$. Hence, so is $\mathbf{G}$. As $\mathbf{G}$ is a stochastic matrix [16], $\mathbf{G} = (\mathbf{0}, \mathbf{0}, \mathbf{e}, \mathbf{0})$, where $\mathbf{e} = (1, 1, 1, 1)^t$, for any n_1,n_2, and m.

To compute $\mathbf{Q}^{(i)}$, $i = -1, -2, ..., -m$, first compute $\mathbf{Q}^{(i)}$ for $-n_2 \leq i \leq -1$ and $i = -m$ by the following recursive formula. Let $\mathbf{Q}^{(i)} = \mathbf{0}$ for $i < -m$.

1. $\mathbf{Q}_{(1)}^{(i)} = \hat{\mathbf{P}}^{(i)}$. Note $\hat{\mathbf{P}}^{(i)} = \mathbf{0}$ if $i \neq -n_2, -m$ and $i < 0$.

2. $\mathbf{Q}_{(k)}^{(i)} = \hat{\mathbf{P}}^{(i)} + \hat{\mathbf{P}}^{(n_1)} \sum_{\mathbf{l}} \mathbf{Q}_{(k-1)}^{(l_1)} \mathbf{Q}_{(k-1)}^{(l_2)} \cdots \mathbf{Q}_{(k-1)}^{(l_\sigma)}$. The sum is over the index range $-m \leq l_1, l_2, ..., l_\sigma \leq -1$, $l_1 + l_2 + \cdots l_\sigma = i - n_1$, and if $i = -m$, $l_\sigma = -m$, if $i \neq -m$, $-n_2 \leq l_\sigma \leq i$, where σ is any positive integer that yields a valid set of $l_1, l_2, ..., l_\sigma$.

$\mathbf{Q}_{(k)}^{(i)}$ converges to $\mathbf{Q}^{(i)}$ geometrically. Matrices $\mathbf{Q}^{(i)}$, $-m < i < -n_2$ can then be computed by the following iterative procedure. Let $\mathbf{R} = \mathbf{I} - \hat{\mathbf{P}}^{(n_1)} \sum_{\mathbf{l}} \mathbf{Q}^{(l_1)} \mathbf{Q}^{(l_2)} \cdots \mathbf{Q}^{(l_\eta)}$. The sum is over the index range $-n_1 \leq l_1, l_2, ..., l_\eta \leq -1$ and $l_1 + l_2 + \cdots l_\eta = n_1$, where η is any positive integer that yields a valid set of l_1, l_2,...,l_η.

1. Set $i = -m + 1$.

2. $\mathbf{Q}^{(i)} = \mathbf{R}^{-1} \hat{\mathbf{P}}^{(n_1)} \sum_{\mathbf{l}} \mathbf{Q}^{(l_1)} \mathbf{Q}^{(l_2)} \cdots \mathbf{Q}^{(l_\sigma)}$. The sum is over the index range $i - n_1 \leq l_\sigma \leq i - 1$, $l_1, l_2, ..., l_{\sigma-1} \leq -1$, and $l_1 + l_2 + \cdots l_\sigma = i - n_1$.

3. Set $i + 1 \rightarrow i$. If $i < -n_2$, go back to step 2; otherwise, stop.

Compute $\mathbf{Q} = \sum_{i=-m}^{-1} \mathbf{Q}^{(i)}$.

References

[1] Altschul, S., W. Gish, W. Miller, E. Myers and D. Lipman (1990) Basic local alignment search tool. *J. Mol. Biol.* **215**, 403-417.

[2] Baum, L. E. and T. Petrie (1966) Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, **37**, 1554-1563.

[3] Baum, L. E., T. Petrie, G. Soules, and N. Weiss (1970) A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, **41**, 164-171.

[4] Bernardi, G., B. Olofsson, J. Filipski, M. Zerial, J. Salinas, G. Cuny, M. Meunier-Rotival, F. Rodier (1985) The mosaic genome of warm-blooded vertebrates. *Science* **228**, 953-958.

[5] Bernardi, G. (2000) Isochores and the evolutionary genomics of vertebrates. *Gene* **241**, 3-17.

[6] Bohr, V. A., D. Phillips and P. C. Hanawalt (1987) Heterogeneous DNA damage and repair in the mammalian genome. *Cancer Research* **47**, 6426-6436.

[7] Boulikas, T. (1992) Evolutionary consequences of nonrandom damage and repair of chromatin domains. *J. Mol. Evol.* **35**, 156-180.

[8] DeSilva, U., L. Elnitski, J. Idol, J. Doyle, W. Gan, J. Thomas, S. Schwartz, N. Dietrich, W. Beckstrom-Sternberg, J. McDowell, R. Blakesley, G. Bouffard, P. Thomas, J. Touchman, W. Miller and E. D. Green (2001) Generation and comparative analysis of ∼3.3 Mb of mouse genomic sequence orthologous to the region of human chromosome 7q11.23 implicated in Williams Syndrome. Submitted.

[9] Durbin, R., S. Eddy, A. Krogh, and G. Mitchison (1998) *Biological Sequence Analysis—Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press.

[10] Endrizzi, M., S. Huang, J. J. Scharf, A.-R. Kelter, B, Wirth, L. M. Kunkel, W. Miller and W. F. Dietrich (1999) Comparative sequence analysis of the mouse and human *Lgn1*/SMA interval. *Genomics* **60**, 137-151.

[11] Gumucio, D., D. Shelton, K. Blanchard-McQuate, T. Gray, S. Tarle, H. Heilstedt-Williamson, J. Slightom, F. Collins and M. Goodman (1994) Differential phylogenetic footprinting as a means to identify base changes responsible for recruitment of the anthropoid γ gene to a fetal expression pattern. *J. Biol. Chem* **269**, 15371-15380.

[12] Hardison, R., J. Oeltjen and W. Miller (1997) Long human-mouse sequence alignments reveal novel regulatory elements: a reason to sequence the mouse genome. *Genome Research* **7**, 959-966.

[13] Huang, X., P. Pevzner and W. Miller (1994) Parametric recomputing in alignment graphs. *Combinatorial Pattern Matching '94*, Lecture Notes in Computer Science 807, Springer-Verlag, pp. 87-101.

[14] Jang, W., A. Hua, S. V. Spilson, W. Miller, B. A. Roe and M. H. Meisler (1999) Comparative sequence of human and mouse BAC clones from the *mnd2* region of chromosome 2p13. *Genome Research* **9**, 53-61.

[15] Karlin, S., and S. F. Altschul (1990) Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl. Acad. Sci. USA* **87**, 2264-2268.

[16] Karlin, S., and A. Dembo (1992) Limit distributions of maximal segmental score among Markov-dependent partial sums. *Adv. Appl. Prob.* **24**, 113-140.

[17] Koop, B. F. (1995) Human and rodent DNA sequence comparisons: a mosaic model of genomic evolution. *Trends in Genetics* **11**, 367-371.

[18] Lund, J, F. Chen, A. Hua, B. Roe, M. Budarf, B. Emanuel and R. H. Reeves (2000) Comparative sequence analysis of 634 kb of the mouse chromosome 16 region of conserved synteny with the human velocardiofacial syndrome region on Chromosome 22q11.2. *Genomics* **63**, 374-383.

[19] Matassi, G., P. M. Sharp and C. Gautier (1999) Chromosomal location effects on gene sequence evolution in mammals. *Current Biology* **9**, 786-791.

[20] Schwartz, S., Z. Zhang, K. A. Frazer, A. Smit, C. Riemer, J. Bouck, R. Gibbs, R. Hardison and W. Miller (2000) PipMaker — a Web server for aligning two genomic DNA sequences. *Genome Research* **10**, 577-586.

[21] Schwarz, G. (1978) Estimating the dimension of a model. *Annals of Statistics*, **6**, 461-464.

[22] Wolfe, K. H., P. M. Sharp and W.-H. Li (1989) Mutation rates differ among regions of the mammalian genome. *Nature* **337**, 283-285.