The statistics of k-mers from a sequence undergoing a simple mutation process without spurious matches*†

Antonio Blanca[1] Robert S. Harris[2] David Koslicki[1,2,3] Paul Medvedev[1,3,4,‡]

[1] Department of Computer Science and Engineering, The Pennsylvania State University
[2] Department of Biology, The Pennsylvania State University
[3] Huck Institutes of the Life Sciences, The Pennsylvania State University
[4] Department of Biochemistry and Molecular Biology, The Pennsylvania State University
‡ Corresponding author, pzm11@psu.edu

Abstract

K-mer-based methods are widely used in bioinformatics, but there are many gaps in our understanding of their statistical properties. Here, we consider the simple model where a sequence S (e.g. a genome or a read) undergoes a simple mutation process whereby each nucleotide is mutated independently with some probability r, under the assumption that there are no spurious k-mer matches. How does this process affect the k-mers of S? We derive the expectation and variance of the number of mutated k-mers and of the number of islands (a maximal interval of mutated k-mers) and oceans (a maximal interval of non-mutated k-mers). We then derive hypothesis tests and confidence intervals for r given an observed number of mutated k-mers, or, alternatively, given the Jaccard similarity (with or without minhash). We demonstrate the usefulness of our results using a few select applications: obtaining a confidence interval to supplement the Mash distance point estimate, filtering out reads during alignment by Minimap2, and rating long read alignments to a de Bruijn graph by Jabba.

*Authors are listed in alphabetical order

†This is the full version of the paper of the same title appearing in the proceedings of RECOMB 2021.

1 Introduction

K-mer-based methods have become widely used, e.g. for genome assembly [1], error correction [26], read mapping [16, 14], variant calling [31], genotyping [32, 7], database search [29, 11], metagenomic sequence comparison [37], and alignment-free sequence comparison [30, 22, 27]. A simple but influential recent example has been the Mash distance [22], which uses the minhash Jaccard similarity between the sets of k-mers in two sequences to estimate their average nucleotide divergence. Mash has been applied to determine the appropriate reference genome for in silico analyses [28], for genome compression [33], for clustering genomes [22, 3], and for estimating evolutionary distance from low-coverage sequencing datasets [27]. K-mer-based methods such as Mash are often faster and more practical then alignment-based methods. However, while the statistics behind sequence alignment are well understood [10], there are many gaps in our understanding of the statistics behind k-mer-based methods.

Consider the following simple mutation model and the questions it raises. There is a sequence of nucleotides S that undergoes a mutation process, whereby every position is mutated with some constant probability r_1, independently of other nucleotides. In this model, we assume that S does not have any repetitive k-mers and that a mutation always results in a unique k-mer (we say that there are no *spurious matches*). This mutation model captures both a simple model of sequence evolution (e.g. Jukes-Cantor) and a simple model of errors generated during sequencing, under the assumptions that k is large enough and the repeat content low enough to make the effect of spurious matches negligible. It is applied to analyze algorithms and the predictions of the model often reflect performance on real biological sequences (e.g. [27, 22]).

How does this simple mutation model affect the k-mers of S? This question bears resemblance but is distinct from questions studied by Lander and Waterman [15] and in alignment-free sequence comparison [30] (we elaborate on the connection in Section 1.1). Some aspects of this question have been previously explored (e.g. [19, 26, 34]), but some very basic ones have not. For example, what is the distribution of the number of mutated k-mers? The expectation of this distribution is known and trivial to derive, but we do not know its variance. For another example, consider that the k-mers of S fall into mutated stretches (which, inspired by Lander-Waterman statistics, we call islands) and non-mutated stretches (which we call oceans). What is the distribution on the number of these stretches? We do not even know the expected value. We answer these and other questions in this paper, with most of the results captured in Table 1.

We immediately apply our findings to derive hypothesis tests and confidence intervals for r_1 from the number of observed mutated k-mers, the Jaccard similarity, and the Jaccard similarity under minhash. Previously, none were known, even though point estimates from these had been frequently used (e.g. Mash). In order to do this, we observe that our random variables are m-dependent [13], which, roughly speaking, means that the only dependencies involve k-mers nearby in the sequence. We apply a technique called Stein's method [25] to approximate these as Normal variables and thereby obtain hypothesis tests and confidence intervals.

We demonstrate the usefulness of our results using a few select applications: obtaining a confidence interval to supplement the Mash distance point estimate [22], filtering out reads during alignment by Minimap2 [16], and rating long read alignments to a de Bruijn graph by Jabba [19]. These examples illustrate how the use of the simple mutation model and the techniques from our paper could have potentially improved several widely used tools. Our technique can also be applied to new questions as they arise. Our code for computing all the intervals in this paper is freely available at `https://github.com/medvedevgroup/mutation-rate-intervals`.

1.1 Related work

Here we give more background on how our paper relates to other previous work.

Lander-Waterman statistics: There is a natural analogy between the stretches of mutated k-mers and the intervals covered by random clones in the work of Lander and Waterman [15]. Each error can be viewed as a random clone with fixed length k, and thus the islands in our study correspond to "covered islands" in theirs. However, their focus was to determine how much redundancy was necessary to cover all (or most) of a genomic sequence, which would correspond to how many nucleotide mutations are needed so that most of the k-mers in the sequence are mutated. In particular, they expect average coverage of the sequence by clones to be greater than 1, while in our study we expect the corresponding value, $\approx k(1-(1-r)^k$, to be much less than 1. Thus, the approximations applied in [15] do not hold in our case.

Alignment-free sequence comparison: In alignment-free analysis, two sequences are compared by comparing their respective vectors of k-mer counts [30]. Two such vectors can be compared in numerous ways, e.g. through the the D_2 similarity measure, which can be viewed as a generalization of the number of mutated k-mers we study in this paper. However, in alignment-free analysis, both the underlying model and the questions studied are somewhat different. In particular, alignment-free analysis usually works with much smaller values of k, e.g. $k < 10$ [38]. This means that most k-mers are present in a sequence, and k-mers will match between and within sequences even if they are in different locations and not evolutionarily related. Our model and questions assume that these spurious matches are background noise that can be ignored (which is justifiable for larger k), while they form a crucial component of alignment-free analysis. As a result, much of the work in measuring expectation and variance in metrics such as D_2 is done with respect to the distribution of the original sequences, rather than after a mutation process [23, 5]. Even when the mutation processes have been studied, they have typically been very different from the ones we consider here (e.g. the "common motif model" [23]). Later works [20, 24] did consider the simple mutation model that we study here, though still with a small k. Sequence similarity has also been estimated using the average common substring length between two sequences [12]. This is similar to the distribution of oceans that we study in our paper, but the difference is that oceans are both left- and right-maximal, while the common substrings considered by [12] and others are only right-maximal.

2 Preliminaries

Let $L > 0$ be a positive integer. Let $[L]$ to denote the interval of integers $\{0, \ldots, L-1\}$, which intuitively captures positions along a string. Let $k > 0$ be a positive integer. The *k-span* at position $0 \leq i < L$ is denoted as K_i and is the range of integers $[i, i+k-1]$ (inclusive of the endpoints). Intuitively, a k-span captures the interval of a k-mer. We think of $[L+k-1]$ as representing an interval of length $L+k-1$ that contains L k-spans. To simplify the statements of the theorems, we will in some places require that $L \geq k$ (or similar), i.e. that the interval is of length at least $2k-1$. We believe this covers most practical cases of interest, but, if necessary, the results can be rederived without this assumption.

We define the *simple mutation model* as a random process that takes as input two integers $k > 0$ and $L > 0$ and a real-valued *nucleotide error rate* $0 < r_1 < 1$. For every position in $[L+k-1]$, the process *mutates* it with probability r_1. A mutation at position i is said to *mutate* the k-spans

Variable	Expectation	Variance	$(1-\alpha)$ interval
N_{mut}	Lq	$L(1-q)(q(2k+\frac{2}{r_1}-1)-2k)+f(r_1,k)$	$Lq \pm z_\alpha\sqrt{\text{Var}(N_{\text{mut}})}$
N_{isl}	$Lr_1(1-q)+f(r_1,k)$	$Lr_1(1-q)(1-r_1(1-q)(2k+1))+f(r_1,k)$	$\text{E}[N_{\text{isl}}] \pm z_\alpha\sqrt{\text{Var}(N_{\text{isl}})}$
N_{ocean}	$Lr_1(1-q)+f(r_1,k)$	$Lr_1(1-q)(1-r_1(1-q)(2k+1))+f(r_1,k)$	$\text{E}[N_{\text{ocean}}] \pm z_\alpha\sqrt{\text{Var}(N_{\text{ocean}})}$
Jaccard	—	—	$\left(\frac{L-Lq-z_\alpha\sqrt{\text{Var}(N_{\text{mut}})}}{L+Lq+z_\alpha\sqrt{\text{Var}(N_{\text{mut}})}}, \frac{L-Lq+z_\alpha\sqrt{\text{Var}(N_{\text{mut}})}}{L+Lq-z_\alpha\sqrt{\text{Var}(N_{\text{mut}})}}\right)$
minhash Jaccard	—	—	see Theorem 6
C_{ber}	$\frac{L(1-q)(1+r_1(k-1))+f(r_1,k)}{L+k-1}$	see Theorem 11	$\text{E}[C_{\text{ber}}] \pm z_\alpha\sqrt{\text{Var}(C_{\text{ber}})}$

Table 1: The expectation, variances, and hypothesis tests derived in this paper. We use q as shorthand for $1-(1-r_1)^k$. We use $f(r_1,k)$ as a placeholder for some function of r_1 and k that is independent of L; see the theorems for the full expressions.

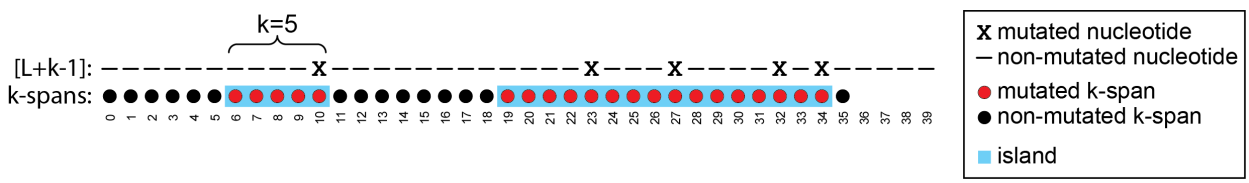

Figure 1: An example of the simple mutation process, with $L = 36$ and $k = 5$. There are 5 nucleotides that are mutated (marked with an x). For example, the mutation at position 10 mutates the k-spans $K_6, \ldots, K_{10}$ (marked in red). Note that an isolated nucleotide mutation (e.g. at position 10) can affect up to k k-spans (e.g. $K_6, \ldots, K_{10}$), but nearby nucleotide mutations can affect the same k-span (e.g. mutation of nucleotides at positions 23 and 27 both affect K_{23}.) There are 2 islands (marked in blue) and 3 oceans, and $N_{\text{mut}} = 21$. For example, $K_{19}, \ldots, K_{34}$ is an island, and K_{35} is an ocean.

$K_{\max(0,i-k+1)}, \ldots, K_i$. We define N_{mut} as a random variable which is the number of mutated k-spans. As shorthand notation, we use $q \triangleq 1-(1-r_1)^k$ to denote the probability that a k-span is mutated. Figure 1 shows an example.

The simple mutation model formalizes the notion of a string S undergoing mutations where there are no spurious matches, i.e. there are no duplicate k-mers in S and a mutation always creates a unique k-mer. This is also closely related to assuming that S is random and k is large enough so that such spurious matches happen with low probability. The simple mutation model captures these scenarios by representing S using the interval $[L+k-1]$ and a k-mer as a k-span.

We can partition the sequence $K_0, \ldots, K_{L-1}$ into alternating intervals called *islands* and *oceans*. The range $i, \ldots, j$ is an *island* iff all $K_i, \ldots, K_j$ are mutated, and the range is maximal, i.e. K_{i-1} and K_{j+1} are either not mutated or out of bounds. Similarly, the range is an *ocean* iff none of $K_i, \ldots, K_j$ are mutated, and the interval is maximal. We define N_{ocean} as a random variable which is the number of oceans and N_{isl} as the number of islands (see Figure 1).

Consider two strings composed of a set of k-mers A and B, respectively, and let $s \leq \min|A|, |B|$ be a non-negative integer. The *Jaccard similarity* between A and B is defined as $\frac{|A \cap B|}{|A \cup B|}$. The *minhash sketch* C_S of a set C is the set of the s smallest elements in C, under a uniformly random permutation hash function. The *minhash Jaccard similarity* between A and B is defined as $\frac{|(A \cup B)_S \cap A_S \cap B_S|}{|(A \cup B)_S|}$, or, equivalently, $|(A \cup B)_S \cap A_S \cap B_S|/s$ [2]. In order to transplant this to our model, we define the *sketching simple mutation model* as an extension of the simple mutation model, with an additional non-negative integer parameter $s \leq L$. We follow the intuition of $[L+k-1]$ representing a string S with no spurious matches. For every position i, if K_i is non-mutated (respectively, mutated), we think of K_i as being shared (respectively, distinct) between the strings before and after the mutation process. Formally, let $\mathcal{U}$ be a universe which contains an element $shared_i$ for every non-

mutated K_i and, for every mutated K_i, contains two elements *a-distinct*$_i$ and *b-distinct*$_i$. Let A be the set of all *shared*$_i$ and *a-distinct*$_i$, and let B be the set of all *shared*$_i$ and *b-distinct*$_i$. The output of the sketching simple mutation model is the minhash Jaccard similarity between A and B, i.e. $\hat{J} = |(A \cup B)_S \cap A_S \cap B_S|/s$. Note that the Jaccard similarity (without sketches) would, in our simple mutation model, be the ratio between the number of *shared*$_i$ and the size of $\mathcal{U}$, which is $\frac{L-N_{\text{mut}}}{L+N_{\text{mut}}}$.

Given a distribution with a parameter of interest p, an *approximate* $(1-\alpha)$*-confidence interval* is an interval which contains p with limiting probability $1-\alpha$. Closely related, an *approximate hypothesis test with significance level* $(1-\alpha)$ is an interval that contains a random variable with limiting probability $1-\alpha$. We will drop the word "approximate" in the rest of the paper, for brevity. We will use the notation $X \in x \pm y$ to mean $X \in [x-y, x+y]$. Given $0 < \alpha < 1$, we define $z_\alpha = \Phi^{-1}(1-\alpha/2)$, where Φ^{-1} is the inverse of the cumulative distribution function of the standard Gaussian distribution. Let $H(x, y, z)$ denote the hypergeometric distribution with population size x, y success states in population, and z trials. We define $F_n(a) = \Pr[H(L+n, L-n, s) \geq a]$. Both Φ^{-1} and F_n can be easily evaluated in programming languages such as R or python.

3 Number of mutated k-mers: expectation and variance

In this section, we look at the distribution of N_{mut}, i.e. the number of mutated k-mers. The approach we take to this kind of analysis, which is standard, is to express N_{mut} as a sum of indicator random variables whose pairwise dependence can be derived. Let X_i be the 0/1 random variable corresponding to whether or not the k-span K_i is mutated; i.e., $X_i = 1$ iff at least one of its nucleotides is mutated. Hence, $\Pr[X_i = 1] = 1 - (1-r_1)^k \triangleq q$. We can express $N_{\text{mut}} = \sum X_i$. By linearity of expectation, we have

$$\mathrm{E}[N_{\text{mut}}] = \mathrm{E}\left[\sum X_i\right] = Lq. \tag{1}$$

The key to the computation of variance is the joint probabilities of two k-mers being mutated.

Lemma 1. *Let $0 \leq i < j < L$. Then, X_i and X_j are independent if $j - i \geq k$ and $\Pr[X_i = 1, X_j = 1] = 2q - 1 + (1-q)(1-r_1)^{j-i}$ otherwise.*

Proof. Set $\delta = j - i$. If $\delta \geq k$, then K_i and $K_{i+\delta}$ do not overlap and therefore the variables X_i and $X_{i+\delta}$ are independent. Otherwise, consider three events. E_1 is the event that at least one of $K_i, \ldots, K_{i+\delta-1}$ is mutated. E_2 is the event that none of $K_i, \ldots, K_{i+\delta-1}$ is mutated and one of $K_{i+\delta}, \ldots, K_{i+k-1}$ is mutated. E_3 is the event that none of $K_i, \ldots, K_{i+k-1}$ is mutated. Notice that the three events form a partition of the event space and so we can write $\Pr[X_i = 1, X_j = 1] = \Pr[X_i = 1, X_j = 1 \mid E_1]\Pr[E_1] + \Pr[X_i = 1, X_j = 1 \mid E_2]\Pr[E_2] + \Pr[X_i = 1, X_j = 1 \mid E_3]\Pr[E_3] = \Pr[X_j = 1 \mid E_1]\Pr[E_1] + 1 \cdot \Pr[E_2] + 0 \cdot \Pr[E_3] = q(1-(1-r_1)^\delta) + (1-r_1)^\delta(1-(1-r_1)^{k-\delta}) = q - q(1-r_1)^\delta + (1-r_1)^\delta - (1-q) = 2q - 1 + (1-q)(1-r_1)^\delta$.

□

We can now compute the variance using tedious but straightforward algebraic calculations. As we will show in the following section, knowing the variance allows us to obtain a confidence interval or do a hypothesis test based on N_{mut}.

Theorem 2. *If $L \geq k$, $\mathrm{Var}(N_{mut}) = L(1-q)(q(2k + \frac{2}{r_1} - 1) - 2k) + k(k-1)(1-q)^2 + \frac{2(1-q)}{r_1^2}((1 + (k-1)(1-q))r_1 - q)$.*

4 Hypothesis test for M-dependent variables

Our derivations of hypothesis tests and confidence intervals follows the strategy used for the Binomials, which we now describe so as to provide intuition. In the case of estimating the success probability p of a Binomial variable X when the number of trials L is known, a confidence interval for p is called a binomial proportion confidence interval [6]. There are multiple ways to calculate such an interval, as described and compared in [4], and we will follow the approach of the Wilson score interval [36]. It works by first approximating the Binomial with a Normal distribution and then applying a standard score. The result is that $\Pr\left[|X - Lp| = z_\alpha\sqrt{\mathrm{Var}(X)}\right] = 1 - \alpha + \varepsilon(L,p)$, where $\mathrm{Var}(X) = Lp(1-p)$ and $\varepsilon(L,p)$ is a function such that $\lim_{L\to\infty}\varepsilon(L,p) = 0$; recall that $z_\alpha = \Phi^{-1}(1-\alpha/2)$. This can be solved for X to obtain a hypothesis test $X \in Lp \pm z_\alpha\sqrt{\mathrm{Var}(X)}$. This can be converted into a confidence interval by finding all values of p for which $X \in Lp \pm z_\alpha\sqrt{\mathrm{Var}(X)}$ holds. In the particular case of the Binomial, a closed form solution is possible [36], but, more generally, one can also find the solution numerically.

Though random variables like N_{mut} are not Binomial, they have a specific form of dependence between the trials, which allows us to apply a similar strategy. A sequence of L random variables $X_0, \ldots, X_{L-1}$ is said to be m-dependent if there exists a bounded m (with respect to L) such that if $j - i > m$, then the two sets $\{X_0, \ldots, X_i\}$ and $\{X_j, \ldots, X_{L-1}\}$ are independent [13]. In other words, m-dependence says that the dependence between a sequence of random variables is limited to be within blocks of length m along the sequence. It is known that the sum of m-dependent random variables is asymptotically normal [13] and this was previously used to construct heuristic hypothesis tests and confidence intervals [18]. Even stronger, the rate of convergence of the sum of m-dependent variables to the Normal distribution is known due to a technique called Stein's method (see Theorem 3.5 in [25]). (This technique applies even to the case where m is not bounded, but that will not be the case in our paper.) Here, we apply Stein's method to obtain a formally correct hypothesis test together with a rate of convergence for a sum of m-dependent (not necessarily identically distributed) Bernoulli variables.

Lemma 3. *Let $X = \sum_{i=0}^{L-1} X_i$ be a sum of m-dependent Bernoulli random variables, where X_i has success probability p_i. Let $\mu = \frac{1}{L}\sum_{i=0}^{L-1} p_i$, $0 < \alpha < 1$, and $\sigma_L^2 = \mathrm{Var}(X)$. Then, $\Pr[X \geq L\mu + z_\alpha\sigma_L] = \Pr[X \leq L\mu - z_\alpha\sigma_L] = \alpha/2 - \varepsilon/2$ and*

$$\Pr[X \in L\mu \pm z_\alpha\sigma_L] = 1 - \alpha + \varepsilon,$$

where $|\varepsilon| \leq 2(2/\pi)^{1/4}\sqrt{\frac{m^2}{\sigma_L^3}\sum_{i=0}^{L-1}\mathrm{E}[|X_i|^3] + \frac{\sqrt{28}m^{3/2}}{\sqrt{\pi}\sigma_L^2}\sqrt{\sum_{i=0}^{L-1}\mathrm{E}[X_i^4]}}$.

Proof. Let $Y = (X - L\mu)/\sigma_L$ and let Z be a standard normal random variable. From Theorem 3.6 in [25], we have $d_{\mathrm{W}}(Y,Z) \leq \frac{m^2}{\sigma_L^3}\sum_{i=0}^{L-1}\mathrm{E}[|X_i|^3] + \frac{\sqrt{28}m^{3/2}}{\sqrt{\pi}\sigma_L^2}\sqrt{\sum_{i=0}^{L-1}\mathrm{E}[X_i^4]} \triangleq d_{\max}$, where $d_{\mathrm{W}}(\cdot,\cdot)$ denotes the Wasserstein metric. Since Z is a standard normal random variable, we have the following standard inequality between the Kolmogorov and Wasserstein metrics (see, e.g., Section 3 in [25]):

$$\begin{aligned}\max_a |\Pr[Y \geq a] - \Pr[Z \geq a]| &\leq (2/\pi)^{1/4}\sqrt{d_{\mathrm{W}}(Y,Z)} \\ &\leq (2/\pi)^{1/4} d_{\max} \triangleq \varepsilon_{\max}.\end{aligned}$$

Recall that for a standard normal variable, $\Pr[Z \geq z_\alpha] = \alpha/2$ and so, by the above, $\Pr[Y \geq z_\alpha] \in \alpha/2 \pm \varepsilon_{\max}$. Similarly, since $\Pr[Z \leq -z_\alpha] = \alpha/2$ we obtain $\Pr[Y \leq -z_\alpha] \in \alpha/2 \pm \varepsilon_{\max}$. From the definition of Y it then follows that $\Pr[X \geq L\mu + z_\alpha\sigma_L] \in \alpha/2 \pm \varepsilon_{\max}$ and $\Pr[X \leq L\mu - z_\alpha\sigma_L] \in \alpha/2 \pm \varepsilon_{\max}$, and therefore implies that $\Pr[X \in L\mu \pm z_\alpha\sigma_L] \in 1 - \alpha \pm 2\varepsilon_{\max}$. □

As we will see, m-dependence is well-suited for dealing with variables in the simple mutation model. In most natural cases, the error $|\varepsilon| \to 0$ when $L \to \infty$, and Lemma 3 gives a hypothesis test with significance level $1 - \alpha$.

5 Hypothesis tests for N_{mut} and $\hat{J}$ and confidence intervals for r_1

There is a natural point estimator for r_1 using N_{mut}, defined as $\hat{r}_1 = 1 - (1 - N_{\text{mut}}/L)^{1/k}$. This estimator is both the method of moments and the maximum likelihood estimator, meaning it has nice convergence properties as L increases [35]. In this section, we extend it to a confidence interval and a hypothesis test, both from N_{mut} and $\hat{J}$ (with and without sketching). In the N_{mut} setting, Lemma 1 shows that $X_0, \ldots, X_{L-1}$ are m-dependent with $m = k-1$. Hence we can apply Lemma 3 to $N_{\text{mut}} = \sum_{i=0}^{L-1} X_i$.

Corollary 4. *Let $0 < \alpha < 1$, $n_{low} = Lq - z_\alpha \sqrt{\text{Var}(N_{mut})}$, and $n_{high} = Lq + z_\alpha \sqrt{\text{Var}(N_{mut})}$. Then $\Pr[N_{mut} \geq n_{high}] = \Pr[N_{mut} \leq n_{low}] = \alpha/2 - \epsilon/2$ and*

$$\Pr[n_{low} \leq N_{mut} \leq n_{high}] = 1 - \alpha + \varepsilon,$$

where $|\varepsilon| \leq c/L^{1/4}$ and c is a constant that depends only on r_1 and k. In particular, when r_1 and k are independent of L, we have $\lim_{L\to\infty}(1 - \alpha + \varepsilon) = 1 - \alpha$.

Corollary 4 gives the closed-form boundaries for a hypothesis test on N_{mut}. To compute a confidence interval for q (equivalently, for r_1), we can numerically find the range of q for which the observed N_{mut} lies between n_{low} and n_{high}. In other words, the upper bound on the range would be given by the value of q for which the observed N_{mut} is n_{low} and the lower bound by the value of q for which the observed N_{mut} is n_{high}. These observations are made rigorous in Theorem 5. We will use the notation N^q_{mut} to denote N_{mut} with parameter $r_1 = 1 - (1 - q)^{1/k}$.

Theorem 5. *For fixed k, r_1, and α, for a given observed value of N^q_{mut}, there exists an L large enough such that there exists a unique q_{low} such that $N^q_{mut} = Lq_{low} + z_\alpha \sqrt{\text{Var}(N^{q_{low}}_{mut})}$ and a unique q_{high} such that $N^q_{mut} = Lq_{\text{high}} - z_\alpha \sqrt{\text{Var}(N^{q_{high}}_{mut})}$, and*

$$\Pr[q \in [q_{low}, q_{high}]] = 1 - \alpha + \epsilon,$$

where $|\varepsilon| \leq c/L^{1/4}$ and c is a constant that depends only on r_1 and k. In particular, for fixed r_1 and k, we have $\lim_{L\to\infty}(1 - \alpha + \epsilon) = 1 - \alpha$.

Note that this theorem states that for sufficiently large L, there is a unique solution for the value of q for which the observed N_{mut} is n_{high} (and similarly a unique solution for the value of q for which the observed N_{mut} is n_{low}). For small L, we have no such guarantee (though we believe the theorem holds true for all $L \geq k$); to deal with this possibility, our software verifies if the solutions are indeed unique by computing the derivative inside the proof of Theorem 5 and checking if it is positive. If it is, then the proof guarantees the solutions to be unique; if it is not, our software reports this. However, during our validations, we did not find such a case to occur.

We want to underscore how the difference between a confidence interval and a hypothesis test is relevant in our case. A confidence interval is useful when we have two sequences, one of which having evolved from the other and we would like to estimate their mutation rate from the number of mutated k-spans. A hypothesis test is useful when we know the mutation rate a priori, e.g. the error rate of a sequencing machine. In this case, we may want to know whether a read could have

been generated from a putative genome location, given the number of observed mutated k-spans. We will see both applications in Section 7.

In some cases, N_{mut} is not observed but instead we observe another random variable $T = f(N_{\text{mut}})$, where $f(x)$ is a monotone function. For example, if $f(x) = (L-x)/(L+x)$, then T is the Jaccard similarity between the original and the mutated sequence (in our model). In this case, a hypothesis test with significance level α is to check if T lies between $f(n_{\text{low}})$ and $f(n_{\text{high}})$. In addition to the Jaccard, [17] describe 14 other variables that are a function of N_{mut}, L, and k. These are: Anderberg, Antidice, Dice, Gower, Hamman, Hamming, Kulczynski, Matching, Ochiai, Phi, Russel, Sneath, Tanimoto and Yule. We can apply our hypothesis test to any of these variables, as long as they are monotone with respect to N_{mut}.

We can also use Lemma 3 as a basis for deriving a hypothesis test on $\hat{J}$ in the sketching model. The proof is more involved and interesting in its own right, but is left for the Appendix due to space constraints.

Theorem 6. *Consider the sketching simple mutation model with known parameters s, k, $L \geq k$, r_1, and output $\hat{J}$. Let $0 < \alpha < 1$ and let $m \geq 2$ be an integer. For $0 \leq i \leq m$, let $n_l^i = Lq - z_{i/m}\sqrt{\text{Var}(N_{mut})}$ and $n_h^i = Lq + z_{i/m}\sqrt{\text{Var}(N_{mut})}$. Let*

$$j_{high} = s^{-1} \min \left\{ a \geq 0 : m\alpha > \sum_{i:n_l^i>0} F_{\lceil n_l^i \rceil}(a) + \sum_{i:n_h^i \leq L} F_{\lceil n_h^{i-1} \rceil}(a) \right\}; \text{ and}$$

$$j_{low} = s^{-1} \max \left\{ a \leq s : m(2-\alpha) < \sum_{i:n_l^i>0} F_{\lfloor n_l^{i-1} \rfloor}(a) + \sum_{i:n_h^i \leq L} F_{\lfloor n_h^i \rfloor}(a) \right\}.$$

Then, assuming that r_1 and k are independent of L, and $m = o(L^{1/4})$,

$$\lim_{L\to\infty} \Pr[j_{low} \leq \hat{J} \leq j_{high}] = 1 - \alpha.$$

We can compute a confidence interval for q from $\hat{J}$ in the same manner as with Corollary 4. Let $j_{\text{low}}(q)$ and $j_{\text{high}}(q)$ be defined as in Theorem 6, but explicitly parameterized by the value of q. Then we numerically find the smallest value $0 < q_{\text{low}} < 1$ for which $j_{\text{low}}(q_{\text{low}}) = \hat{J}$ and the largest value $0 < q_{\text{high}} < 1$ for which $j_{\text{high}}(q_{\text{high}}) = \hat{J}$. The following theorem guarantees that $[q_{\text{low}}, q_{\text{high}}]$ is a confidence interval for q.

Theorem 7. *For fixed k, r_1, α, m, and a given observed value of $\hat{J}$, there exists an L large enough such that there exist unique intervals $[q_{low}^-, q_{low}^+]$ and $[q_{high}^-, q_{high}^+]$ such that $q_{high}^+ \geq q_{low}^-$, $j_{low}(\hat{q}) = \hat{J}$ if and only if $\hat{q} \in [q_{low}^-, q_{low}^+]$, and $j_{high}(\hat{q}) = \hat{J}$ if and only if $\hat{q} \in [q_{high}^-, q_{high}^+]$. Moreover, assuming that r_1, k and m are independent of L, we have*

$$\lim_{L\to\infty} \Pr[q \in [q_{low}^-, q_{high}^+]] = 1 - \alpha.$$

6 Number of islands and oceans

In this section, we derive the expectation and variance of N_{isl} and N_{ocean} and the hypothesis test based on them. For N_{isl}, we follow the same strategy as for N_{mut}, namely to express N_{isl} as a sum of indicator random variables whose joint probabilities can be derived. Let us define a *right border* as a position i such that K_i is mutated and K_{i+1} is not. We will denote it by an indicator variable

B_i, for $0 \leq i < L-1$. Let us also say that there exists an *end-of-string border* iff K_{L-1} is mutated. We will denote this by an indicator variable Z. A right border is a position where an island ends and an ocean begins, and the end-of-string border exists if the last island is terminated not by an ocean but by the end of available nucleotides in the string to make a k-mer. The number of islands is then the number of borders, i.e. $N_{\text{isl}} = Z + \sum_{i=0}^{L-2} B_i$.

To compute the expectation, observe that Z is a Bernoulli variable with parameter q. For B_i, observe that the only way that K_i is mutated while K_{i+1} is not is if position i is mutated and the positions $i+1, \ldots, i+k$ are not. Therefore, $B_i \sim Bernoulli(r_1(1-q))$. By linearity of expectation,

$$\mathrm{E}[N_{\text{isl}}] = q + r_1(1-q)(L-1) = Lr_1(1-q) + q - r_1(1-q). \tag{2}$$

Next, we derive dependencies between border variables and use them to compute the variance.

Lemma 8. *Let $0 \leq i < j \leq L-2$. Then $\Pr[B_i = 1, B_j = 1] = 0$ if $j \leq i+k$ and $\Pr[B_i = 1, B_j = 1] = \Pr[B_i = 1]\Pr[B_j = 1] = r_1^2(1-q)^2$ otherwise. Also, $\Pr[B_i = 1, Z = 1] = \Pr[B_i = 1]\Pr[Z = 1] = r_1 q(1-q)$ if $i \leq L-2-k$, and $\Pr[B_i = 1, Z = 1] = r_1(1-q)(1-(1-r_1)^{L-2-i})$ otherwise.*

Proof. Observe that when $j - i > k$, the positions that have an effect on B_i (i.e. $K_i, \ldots, K_{i+k}$) and those that have an effect on B_j (i.e. $K_j, \ldots, K_{j+k}$) are disjoint. Hence, B_i and B_j are independent in this case. When $1 \leq j - i \leq k$, B_i and B_j cannot co-occur. This is because $B_i = 1$ implies that position j is not mutated, while $B_j = 1$ implies that it is. By the same logic, Z is independent of all B_i for $0 \leq i \leq L-2-k$. For the case when $L-2-k < i \leq L-2$, $B_i = 1$ implies that positions $L-1, \ldots, i+k$ are not mutated. Therefore, there is an end-of-string border when $B_i = 1$ iff one of the positions $i+k+1, \ldots, L+k-2$ is mutated. Thus, $\Pr[Z = 1, B_i = 1] = \Pr[Z = 1 \mid B_i = 1]\Pr[B_i = 1] = (1-(1-r_1)^{L+k-2-(i+k+1)+1})r_1(1-q)$. □

Theorem 9. *For $L \geq k+3$, $\mathrm{Var}(N_{isl}) = Lr_1(1-q)(1-r_1(1-q)(2k+1)) + k^2 r_1^2(1-q)^2 + k(r_1(3r_1+2)(1-q)^2) + (1-q)((1-q)r_1^2 - q - r_1)$.*

Lemma 8 also shows that N_{isl} is m-dependent, with $m = k-1$, Therefore, a hypothesis test on N_{isl} can be obtained as a corollary of Lemma 3.

Corollary 10. *Fix r_1 and let $0 < \alpha < 1$. Then, the probability that $N_{isl} \in \mathrm{E}[N_{isl}] \pm z_\alpha \sqrt{\mathrm{Var}(N_{isl})}$ is $1 - \alpha + \varepsilon$, where $|\varepsilon| \leq c/L^{1/4}$ and c is a constant that depends only on r_1 and k. In particular, when r_1 and k are independent of L, we have $\lim_{L\to\infty}(1-\alpha+\varepsilon) = 1-\alpha$.*

Unlike for Corollary 4, it is not as straightforward to invert this hypothesis test into a confidence interval for r_1, since the endpoints of the interval of N_{isl} are not monotone in r_1. We therefore do not pursue this direction here. The derivation of the expectation and variance for N_{ocean} is analogous and left for the Appendix (Theorem 12). Observe that $|N_{\text{ocean}} - N_{\text{isl}}| \leq 1$, so, as expected, the expectation and variance are identical to N_{isl} in the higher order terms. Corollary 10 also holds for the case that n is the observed number of oceans, if we just replace N_{isl} with N_{ocean}.

An immediate application of N_{ocean} is to compute a hypothesis test for the *coverage by exact regions* (C_{ber}), a variable earlier applied to score read mappings in [19]. C_{ber} is the fraction of positions in $[L+k-1]$ that lie in k-spans that are in oceans. The total number of bases in all the oceanic k-spans is the number of non-mutated k-spans plus, for each ocean, an extra $k-1$ "starter" bases. We can then write

$$C_{\text{ber}} = (L - N_{\text{mut}} + (k-1)N_{\text{ocean}})/(L+k-1).$$

We can use the expectations and variances of N_{mut} (eq. (1) and Theorem 2) and N_{ocean} (Theorem 12) to derive the expectation and variance of C_{ber}:

	$L = 100$				$L = 1,000$				$L = 10,000$				*E.Coli*			
$r_1 =$	0.001	0.01	0.1	0.2	0.001	0.01	0.1	0.2	0.001	0.01	0.1	0.2	0.001	0.01	0.1	0.2
$k = 100$	0.91	1.00	NA	NA	0.95	0.96	NA	NA	0.95	0.95	NA	NA	0.95	0.95	NA	NA
$k = 51$	0.91	1.00	1.00	NA	0.94	0.95	0.94	NA	0.95	0.95	0.96	NA	0.95	0.95	0.95	NA
$k = 21$	0.91	0.96	1.00	1.00	0.93	0.95	0.95	0.95	0.95	0.94	0.95	0.95	0.95	0.94	0.93	0.94

Table 2: The accuracy of the confidence intervals for r_1 predicted by Corollary 4, for $\alpha = 0.05$ and for various values of L, r_1, and k (the first three groups) and for the *E.coli* sequence (the fourth group). NA indicates the experiment was not run; for the first three groups, we only ran on parameters where $\lceil \mathrm{E}[N_{\mathrm{mut}}] \rceil < L$ (otherwise they were not of interest), while for *E.coli*, we ran with the same range of values of r_1 and k as in the first three groups. In each cell, we report the fraction of 10,000 replicates for which the true r_1 falls into the predicted confidence interval. For the *E.coli* sequence, we used strain Shigella flexneri Shi06HN159.

Theorem 11. $\mathrm{E}[C_{ber}] = \frac{1-q}{L+k-1}\left(L(1+r_1(k-1)) + (1-r_1)(k-1)\right)$ *and, for* $L \geq k+3$, $\mathrm{Var}(C_{ber}) = \frac{(1-q)(cL+d)}{r^2(L+k-1)}$, *where*

$$c = 2rq + r^2(-3q - 2k + 4kq) + r^3(k-1)(4kq - 3k - 1) + r^4(1-q)(k-1)^2(-2k-1); \text{ and}$$
$$d = -2q + 2r(q + k - kq) + r^2(k-1)(k-q)$$
$$+ r^3(k-1)(3k - 4kq + 1) + r^4(k-1)^2(1-q)(k^2 + 3k + 1).$$

Then, observing that C_{ber} is a linear combination of m-dependent variables and hence itself m-dependent, we can apply Lemma 3 and obtain that, when r_1 and k are independent of L, $\lim_{L\to\infty} \Pr[C_{\mathrm{ber}} \in \mathrm{E}[C_{\mathrm{ber}}] \pm z_\alpha \sqrt{\mathrm{Var}(C_{\mathrm{ber}})}] = 1 - \alpha$.

7 Empirical results and applications

In this section, we evaluate the accuracy of our results and demonstrate several applications. A sanity check validation of the correctness of our formulas for $\mathrm{E}[N_{\mathrm{mut}}]$ and $\mathrm{Var}[N_{\mathrm{mut}}]$ is shown in Table S1, however, most of the expectation and variance formulas are evaluated indirectly through the accuracy of the corresponding confidence intervals. We focus the evaluation on accuracy rather than run time, since calculating the confidence interval took no more than a few seconds for most cases (the only exception was for sketch sizes of 100k or more, the evaluation took on the order of minutes). Memory use was negligible in all cases.

7.1 Confidence intervals based on N_{mut}

In this section, we evaluate the accuracy of the confidence intervals (CIs) produced by Corollary 4 (other CIs will be evaluated indirectly through applications). We first simulate the simple mutation model to measure the accuracy, shown in the left three groups (i.e. $L = 100, 1000, 10000$) of Table 2, for $\alpha = 0.05$. We observe that the predicted CIs are very accurate at $L = 1000$, and also accurate for smaller k and r_1 when $L = 100$. Similar results hold for $\alpha = 0.01$ (Table S2) and $\alpha = 0.10$ (Table S3). The remainder of the cases had a CI that was too conservative; these are also the cases with some of the smallest variances (Table S1) and we suspect that, similar to the case of the Binomial, the Normal approximation of m-dependent variables deteriorates with very small variances. However, further investigation is needed.

Next, we investigate how well our predictions hold up when we simulate mutations along a real genome, where we can only observe the set of k-mers without their positions in the genome (as in alignment-free sequence comparison). We start with the *E.coli* genome sequence and, with probability r_1, for every position, flip the nucleotide to one of three other nucleotides, chosen with

Sketch size	$r_1 = .05$, $q = .659$			$r_1 = .15$, $q = .967$			$r_1 = .25$, $q = .998$		
	accuracy	low	high	accuracy	low	high	accuracy	low	high
100	0.97	0.037	0.069	1.00	0.103	0.303	1.00	0.119	1.000
1,000	0.96	0.046	0.055	0.97	0.133	0.174	1.00	0.193	0.375
10,000	0.95	0.049	0.051	0.96	0.144	0.156	0.96	0.232	0.277
100,000	0.95	0.049	0.051	0.95	0.148	0.152	0.96	0.243	0.257
1,000,000	0.94	0.050	0.050	0.95	0.149	0.151	0.96	0.247	0.253

Table 3: The confidence intervals predicted by Theorem 6 and their accuracy. For each sketch size and r_1 value, we show the number of trials for which the true r_1 falls within the predicted confidence interval. The reported CI corresponds to applying Theorem 6 with $\hat{J} = \frac{1-q}{1+q}$. Here, $\alpha = 0.05$, $k = 21$, $L = 4,500,000$, and the sketch size s and r_1 are varied as shown. The number of trials for each cell is 1,000, and $m = 100$ for Theorem 6.

equal probability. Let A and B be the set of distinct k-mers in *E.coli* before and after the mutation process, respectively. We let $L = (|A| + |B|)/2$ and $n = L - |A \cap B|$. We then calculate the 95% CI for r_1 under the simple mutation model (Corollary 4) by plugging in n for N_{mut}. The rightmost group in Table 2 shows the accuracy of these CIs. We see that the simple mutation model we consider in this paper is a good approximation to mutations along a real genome like *E.coli*.

7.2 Mash distance

The Mash distance [22] (and its follow-up modifications [21, 27]) first measures the minhash Jaccard similarity j between two sequences and then uses a formula to give a point estimate for r_1 under the assumptions of the sketching simple mutation model. While a hypothesis test was described in [22], it was only for the null model where the two sequences were unrelated. Theorem 6 allows us instead to give a CI for r_1, based on the minhash Jaccard similarity, in the sketching simple mutation model. Table 3 reproduces a subset of Table 1 from [22], but using CIs given by Theorem 6. For most cases, the predicted CIs are highly accurate, with an error of at most two percentage points. The three exceptions happen when s is small and q is large; in such cases, the predicted CI is too conservative (i.e. too large). In Table S4, we also tested the accuracy with a real *E.coli* genome by letting A and B be the set of distinct k-mers in the genome before and after mutations, respectively, letting $L = (|A| + |B|)/2$ and $\hat{J} = ((|A \cup B|) \cap A_s \cap B_s)/s$, and applying Theorem 6 with those values. The accuracy is very similar to that in the simple mutation model, demonstrating that for a genome like *E.Coli*, the simple mutation model is a good approximation.

7.3 Filtering out reads during alignment to a reference

Minimap2 is a widely used long read aligner [16]. The algorithm first picks certain k-mers in a read as *seeds*. Then, it identifies a region of the read and a region of the reference that potentially generated it (called a chain in [16]). Let n be the number of seeds in the read and let $m \leq n$ be the number of those that exist in the reference region. Minimap2 models the error rate of the k-mers as a homogenous Poisson process and estimates the sequence divergence between the read and the reference as $\hat{\epsilon} = \frac{1}{k} \log \frac{n}{m}$ (which is the maximum likelihood estimator in that model). If $\hat{\epsilon}$ is above a threshold, the alignment is abandoned. [16] observes that due to invalid assumptions, $\hat{\epsilon}$ is only approximate and can be biased, but nevertheless maintains a good correlation with the true divergence.

Using our paper, we can obtain a more accurate estimate of r_1. The situation is very similar to estimating r_1 from N_{mut}, except that only a subset of k-spans are being "tracked." Therefore, the maximum likelihood estimator for q is m/n and for r_1 is $\hat{r}_1 = 1 - (m/n)^{1/k}$. Figures 2 and S3

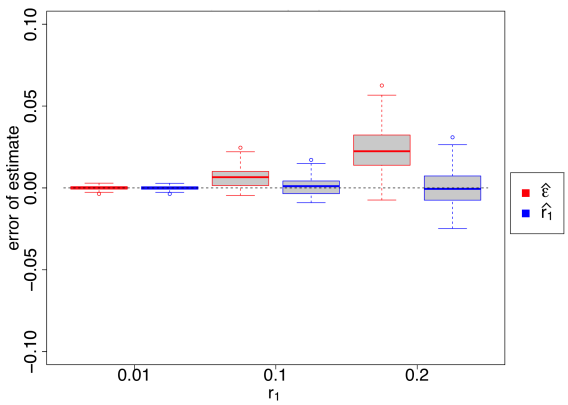

Figure 2: Estimates of sequence divergence as done by mimimap2 ($\hat{\epsilon}$) and by us ($\hat{r}_1$). Reads are simulated from a random 10kbp sequence introducing mutations at the given r_1 rate. For each r_1 value, 100 reads are used. As in [16], we use $k = 15$ and, using a random hash function, identify as seeds the k-mer minimizers, one for every window of 25 k-mers. In the case when $\hat{\epsilon}$ is undefined, we set $\hat{\epsilon} = 1$.

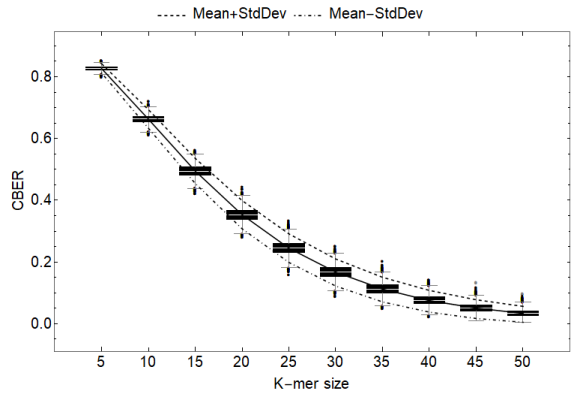

Figure 3: Box and whisker plot of C_{ber} scores for 5,000 replicates of random strings of length 10,000nt, with mutations introduced at a rate of $r_1 = 0.1$. The solid black line corresponds to the empirical median of C_{ber}, while the dashed top line corresponds to $\text{E}[C_{\text{ber}}] + z_{0.05}\sqrt{\text{Var}(C_{\text{ber}})}$ and the bottom dot-dashed line corresponds to $\text{E}[C_{\text{ber}}] - z_{0.05}\sqrt{\text{Var}(C_{\text{ber}})}$, both computed from Theorem 11.

show the relative performance of the two estimators ($\hat{\epsilon}$ and $\hat{r}_1$) for sequences of different lengths, with our $\hat{r}_1$ much closer to the simulated rate than $\hat{\epsilon}$ in both cases.

7.4 Evaluating an alignment of a long read to a graph

Jabba [19] is an error-correction algorithm for long read data. At one stage, the algorithm evaluates whether a read is likely to have originated from a given location in the reference. Because Jabba's reference is a de Bruijn graph and not a string, it uses the specialized C_{ber} score for the evaluation. In this scenario, the mutation process corresponds to sequencing errors at a known error rate r_1 and the question is whether the read is likely to have arisen through this process from the given location of the reference. The authors assume the simple mutation model and derive the expected C_{ber} score as $1 - r_1 - \sum_{i=0}^{k-1} i(1-r_1)^i r_1^2$. They then give a lower rating to reads with a C_{ber} score that has "significant deviation" from this expected value. It is not clear how much of a deviation is deemed to be significant or how it was calculated.

Theorem 11, which gives $\text{E}[C_{\text{ber}}]$ and $\text{Var}[C_{\text{ber}}]$, would have allowed [19] to take a more rigorous approach. It shows that the C_{ber} expectation computed by [19] is correct only in the limit as $L \to \infty$, while our formula is exact and closed-form. More substantially, we can make the determination of "significant deviation" more rigorous. We regenerated Figure 2 from [19], using the same range of values for k (called m in [19]) and an error rate of $r_1 = 10\%$ as in [19] and plotted the 95% confidence interval: $E[C_{\text{ber}}] \pm z_{.05}\sqrt{\text{Var}(C_{\text{ber}})}$. Figure 3 demonstrates that this range would have done a good job at capturing most of the generated reads. Table 4 gives the number of C_{ber} values that fall inside of the 95% confidence interval when using a simple mutation process with the same $r_1 = 10\%$ for sequences of length 10,000 for 5,000 replicates, with k ranging from 5 to 50 in steps of 5, depicting good agreement between simulation and theorem 11.

k-mer size	5	10	15	20	25	30	35	40	45	50
% inside CI	0.95	0.95	0.95	0.95	0.95	0.95	0.94	0.94	0.95	0.93

Table 4: A total of 5,000 sequences, each of length 10,000nt underwent a simple mutation process with mutation probability $r_1 = 0.1$. The percent of associated C_{ber} scores that fell inside of the 95% confidence interval as determined by Theorem 11 are shown.

8 Conclusion

The simple mutation model has been used broadly to model either biological mutations or sequencing errors. However, its use has usually been limited to derive the expectations of random variables, e.g. the expected number of mutated k-mers. In this paper, we take this a step further and show that the dependencies between indicator variables in this model (e.g. whether a k-mer at a given position is mutated) are often easy to derive and are limited to nearby locations. This limited dependency allows us to show that the sum of these indicators is approximately Normal. As a result, we are able to obtain hypothesis tests and confidence tests in this model.

The most immediate application of our paper is likely to compute a confidence interval for average nucleotide identity from the minhash sketching Jaccard. Previously, only a point estimate was available, using Mash. However, we hope that our technique can be applied by others to random variables that we did not consider. All that is needed is to derive the joint probability of the indicator variables and compute the variance. Computing the variance by hand is tedious and error-prone but can be done with the aid of a software like Mathematica.

We test the robustness of the simple mutation model in the presence of spurious matches by using a real *E.coli* sequence. However, we do not explore the robustness with respect to violations such as the presence of indels (which result in different string lengths) or the presence of more repeats than in *E.coli*. This type of robustness has already been explored in other papers that use the simple mutation model [8, 22, 27]. However, exploring the robustness of our confidence intervals in downstream applications is important future work.

On a more technical note, it would be interesting to derive more tight error bounds for our confidence intervals, both in terms of more tightly capturing the dependencies on L, r_1, and k, and accurately tracking constants. The error bound ε that is stated in Lemma 3 is likely not tight in either respect, due to the inherent loss when transferring between the Wasserstein and Kolmogorov metrics and due to loose inequalities within the proof of Theorem 3.5 in [25]. Ideally, tight error bounds would give the user a way to know, without simulations, when the confidence intervals are accurate, in the same way that we know that the Wilson score interval for a Binomial will be inaccurate when $np(1-p)$ is low. For example, it would be useful to better theoretically explain and predict which values in Table 2 deviate from 0.95.

Another practical issue is with the implementation of the algorithm to compute a confidence interval for q from $\hat{J}$. Theorem 7 guarantees that the algorithm is correct as L goes to infinity. However, the user of the algorithm will not know if L is large enough for the confidence interval to be correct. There are several heuristic ways to check this, which we have implemented in the software: a short simulation to check the true coverage of the reported confidence interval, a check that the sets in the definitions of j_{high} and j_{low} are not empty, and a check that j_{high} and j_{low} are monotonic with respect to q in the range $0 < q < 1$.

References

[1] Anton Bankevich, Sergey Nurk, Dmitry Antipov, Alexey A Gurevich, Mikhail Dvorkin, Alexander S Kulikov, Valery M Lesin, Sergey I Nikolenko, Son Pham, Andrey D Prjibelski, et al. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *Journal of computational biology*, 19(5):455–477, 2012.

[2] Andrei Z Broder. On the resemblance and containment of documents. In *Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171)*, pages 21–29. IEEE, 1997.

[3] Christopher T Brown, Matthew R Olm, Brian C Thomas, and Jillian F Banfield. Measurement of bacterial replication rates in microbial communities. *Nature biotechnology*, 34(12):1256, 2016.

[4] Lawrence D Brown, T Tony Cai, and Anirban DasGupta. Interval estimation for a binomial proportion. *Statistical science*, pages 101–117, 2001.

[5] Conrad J Burden, Paul Leopardi, and Sylvain Forêt. The distribution of word matches between markovian sequences with periodic boundary conditions. *Journal of Computational Biology*, 21(1):41–63, 2014.

[6] George Casella and Roger L Berger. *Statistical inference*, volume 2. Duxbury Pacific Grove, CA, 2002.

[7] Luca Denti, Marco Previtali, Giulia Bernardini, Alexander Schönhuth, and Paola Bonizzoni. MALVA: genotyping by Mapping-free ALlele detection of known VAriants. *iScience*, 18:20–27, 2019.

[8] Huan Fan, Anthony R Ives, Yann Surget-Groba, and Charles H Cannon. An assembly and alignment-free method of phylogeny reconstruction from next-generation sequencing data. *BMC genomics*, 16(1):522, 2015.

[9] RL Grajam, Donald E Knuth, and Oren Patashnik. Concrete mathematics, a foundation for computer science, 1988.

[10] Dan Gusfield. *Algorithms on strings, trees and sequences: computer science and computational biology*. Cambridge University Press, 1997.

[11] R. S. Harris and P. Medvedev. Improved Representation of Sequence Bloom Trees. *bioRxiv*, 2018.

[12] Bernhard Haubold, Peter Pfaffelhuber, Mirjana Domazet-Loso, and Thomas Wiehe. Estimating mutation distances from unaligned genomes. *Journal of Computational Biology*, 16(10):1487–1500, 2009.

[13] Wassily Hoeffding, Herbert Robbins, et al. The central limit theorem for dependent random variables. *Duke Mathematical Journal*, 15(3):773–780, 1948.

[14] Chirag Jain, Alexander Dilthey, Sergey Koren, Srinivas Aluru, and Adam M Phillippy. A fast approximate algorithm for mapping long reads to large reference databases. In *International Conference on Research in Computational Molecular Biology*, pages 66–81. Springer, 2017.

[15] E. S. Lander and M. S. Waterman. Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics*, 2(3):231–239, 1988.

[16] Heng Li. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, 34(18):3094–3100, 2018.

[17] Yang Young Lu, Kujin Tang, Jie Ren, Jed A Fuhrman, Michael S Waterman, and Fengzhu Sun. Cafe: accelerated alignment-free sequence analysis. *Nucleic acids research*, 45(W1):W554–W559, 2017.

[18] Weiwen Miao and Joseph L Gastwirth. The effect of dependence on confidence intervals for a population proportion. *The American Statistician*, 58(2):124–130, 2004.

[19] Giles Miclotte, Mahdi Heydari, Piet Demeester, Stephane Rombauts, Yves Van de Peer, Pieter Audenaert, and Jan Fostier. Jabba: hybrid error correction for long sequencing reads. *Algorithms for Molecular Biology*, 11(1):1–12, 2016.

[20] Burkhard Morgenstern, Bingyao Zhu, Sebastian Horwege, and Chris André Leimeister. Estimating evolutionary distances between genomic sequences from spaced-word matches. *Algorithms for Molecular Biology*, 10(1):5, 2015.

[21] Brian D Ondov, Gabriel J Starrett, Anna Sappington, Aleksandra Kostic, Sergey Koren, Christopher B Buck, and Adam M Phillippy. Mash Screen: High-throughput sequence containment estimation for genome discovery. *Genome biology*, 20(1):232, 2019.

[22] Brian D Ondov, Todd J Treangen, Páll Melsted, Adam B Mallonee, Nicholas H Bergman, Sergey Koren, and Adam M Phillippy. Mash: fast genome and metagenome distance estimation using minhash. *Genome biology*, 17(1):132, 2016.

[23] Gesine Reinert, David Chew, Fengzhu Sun, and Michael S Waterman. Alignment-free sequence comparison (i): statistics and power. *Journal of Computational Biology*, 16(12):1615–1634, 2009.

[24] Sophie Röhling, Alexander Linne, Jendrik Schellhorn, Morteza Hosseini, Thomas Dencker, and Burkhard Morgenstern. The number of k-mer matches between two dna sequences as a function of k and applications to estimate phylogenetic distances. *Plos one*, 15(2):e0228070, 2020.

[25] Nathan Ross. Fundamentals of Stein's method. *Probability Surveys*, 8:210–293, 2011.

[26] Leena Salmela, Riku Walve, Eric Rivals, and Esko Ukkonen. Accurate self-correction of errors in long reads using de Bruijn graphs. *Bioinformatics*, 33(6):799–806, 2017.

[27] Shahab Sarmashghi, Kristine Bohmann, M Thomas P Gilbert, Vineet Bafna, and Siavash Mirarab. Skmer: assembly-free and alignment-free sample identification using genome skims. *Genome biology*, 20(1):1–20, 2019.

[28] Oliver Schwengers, Torsten Hain, Trinad Chakraborty, and Alexander Goesmann. Reference-seeker: rapid determination of appropriate reference genomes. *BioRxiv*, page 863621, 2019.

[29] B. Solomon and C. Kingsford. Fast search of thousands of short-read sequencing experiments. *Nature biotechnology*, 34(3):300–302, 2016.

[30] Kai Song, Jie Ren, Gesine Reinert, Minghua Deng, Michael S Waterman, and Fengzhu Sun. New developments of alignment-free sequence comparison: measures, statistics and next-generation sequencing. *Briefings in bioinformatics*, 15(3):343–353, 2014.

[31] Daniel S Standage, C Titus Brown, and Fereydoun Hormozdiari. Kevlar: a mapping-free framework for accurate discovery of de novo variants. *bioRxiv*, page 549154, 2019.

[32] Chen Sun and Paul Medvedev. Toward fast and accurate snp genotyping from whole genome sequencing data for bedside diagnostics. *Bioinformatics*, 35(3):415–420, 2018.

[33] Tao Tang, Yuansheng Liu, Buzhong Zhang, Benyue Su, and Jinyan Li. Sketch distance-based clustering of chromosomes for large genome database compression. *BMC genomics*, 20(10):1–9, 2019.

[34] Anqi Wang and Kin Fai Au. Performance difference of graph-based and alignment-based hybrid error correction methods for error-prone long reads. *Genome biology*, 21(1):14, 2020.

[35] Larry Wasserman. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.

[36] Edwin B Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927.

[37] Derrick E Wood and Steven L Salzberg. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome biology*, 15(3):R46, 2014.

[38] Tiee-Jian Wu, Ying-Hsueh Huang, and Lung-An Li. Optimal word sizes for dissimilarity measures and estimation of the degree of dissimilarity between dna sequences. *Bioinformatics*, 21(22):4125–4132, 2005.

A Appendix

A.1 Missing theorems and proofs

Theorem 2. *If* $L \geq k$, $\mathrm{Var}(N_{mut}) = L(1-q)(q(2k+\frac{2}{r_1}-1)-2k)+k(k-1)(1-q)^2+\frac{2(1-q)}{r_1^2}((1+(k-1)(1-q))r_1-q)$.

Proof. In the following we will use Lemma 1 and the equality $\sum_{i=0}^{n} ix^i = \frac{x-(n+1)x^{n+1}+nx^{n+2}}{(1-x)^2}$ for $x \neq 1$ from [9].

$$
\begin{aligned}
\mathrm{Var}[N_{\mathrm{mut}}] &= E[{N_{\mathrm{mut}}}^2] - E[N_{\mathrm{mut}}]^2 \\
&= \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} E[X_iX_j] - L^2q^2 \\
&= \sum_{i=0}^{L-1} E[X_iX_i] + 2\sum_{\delta=1}^{k-1}\sum_{i=0}^{L-1-\delta} E[X_iX_{i+\delta}] + 2\sum_{\delta=k}^{L-1}\sum_{i=0}^{L-1-\delta} E[X_iX_{i+\delta}] - L^2q^2 \\
&= \sum_{i=0}^{L-1} q + 2\sum_{\delta=1}^{k-1}\sum_{i=0}^{L-1-\delta} 2q-1+(1-q)(1-r)^\delta + 2\sum_{\delta=k}^{L-1}\sum_{i=0}^{L-1-\delta} q^2 - L^2q^2 \\
&= Lq + 2\sum_{\delta=1}^{k-1}(L-\delta)(2q-1+(1-q)(1-r)^\delta) + 2\frac{(L+1-k)(L-k)}{2}q^2 - L^2q^2 \\
&= Lq + ((L+1-k)(L-k)-L^2)q^2 \\
&\quad + 2\sum_{\delta=1}^{k-1}(L-\delta)(2q-1+(1-q)(1-r)^\delta) \\
&= Lq + ((L+1-k)(L-k)-L^2)q^2 \\
&\quad + 2\sum_{\delta=1}^{k-1}(L-\delta)(2q-1) \\
&\quad + 2\sum_{\delta=1}^{k-1}(L-\delta)(1-r)^\delta \\
&\quad - 2\sum_{\delta=1}^{k-1}(L-\delta)q(1-r)^\delta \\
&= Lq + ((L+1-k)(L-k)-L^2)q^2 \\
&\quad + 2\sum_{\delta=1}^{k-1}L(2q-1) - 2\sum_{\delta=1}^{k-1}\delta(2q-1) \\
&\quad + 2\sum_{\delta=1}^{k-1}L(1-r)^\delta - 2\sum_{\delta=1}^{k-1}\delta(1-r)^\delta \\
&\quad - 2\sum_{\delta=1}^{k-1}Lq(1-r)^\delta + 2\sum_{\delta=1}^{k-1}\delta q(1-r)^\delta
\end{aligned}
$$

$$
\begin{aligned}
&= Lq + ((L+1-k)(L-k) - L^2)q^2 \\
&\quad + 2(k-1)L(2q-1) - 2(2q-1)\sum_{\delta=1}^{k-1}\delta \\
&\quad + 2L\sum_{\delta=1}^{k-1}(1-r)^\delta - 2\sum_{\delta=1}^{k-1}\delta(1-r)^\delta \\
&\quad - 2Lq\sum_{\delta=1}^{k-1}(1-r)^\delta + 2q\sum_{\delta=1}^{k-1}\delta(1-r)^\delta \\
&= Lq + ((L+1-k)(L-k) - L^2)q^2 + 2(k-1)L(2q-1) \\
&\quad - 2(2q-1)\sum_{\delta=1}^{k-1}\delta \\
&\quad + (2L-2Lq)\sum_{\delta=1}^{k-1}(1-r)^\delta \\
&\quad + (2q-2)\sum_{\delta=1}^{k-1}\delta(1-r)^\delta \\
&= Lq + ((L+1-k)(L-k) - L^2)q^2 + 2(k-1)L(2q-1) \\
&\quad - 2(2q-1)\frac{k(k-1)}{2} \\
&\quad + (2L-2Lq)\frac{q-r}{r} \\
&\quad + (2q-2)\frac{1-r-(1+(k-1)r)(1-q)}{r^2} \\
&= Lq + ((L+1-k)(L-k) - L^2)q^2 + 2(k-1)L(2q-1) \\
&\quad - (2q-1)k(k-1) \\
&\quad + \frac{2L(1-q)q}{r} - 2L(1-q) \\
&\quad + \frac{(2q-2)(1-r-(1+(k-1)r)(1-q))}{r^2} \\
&= k(k-1)(1-q)^2 - L(2k(1-q)+q)(1-q) \\
&\quad + \frac{2L(1-q)q}{r} + \frac{(2q-2)(1-r-(1+(k-1)r)(1-q))}{r^2} \\
&= \frac{2L(1-q)q}{r} - L(2k(1-q)+q)(1-q) \\
&\quad + k(k-1)(1-q)^2 + \frac{(2q-2)(1-r-(1+(k-1)r)(1-q))}{r^2} \\
&= L(1-q)(q(2k+\frac{2}{r}-1)-2k) \\
&\quad + k(k-1)(1-q)^2 + \frac{2(1-q)}{r^2}((1+(k-1)(1-q))r-q).
\end{aligned}
$$

□

Theorem 5. *For fixed k, r_1, and α, for a given observed value of N^q_{mut}, there exists an L large enough such that there exists a unique q_{low} such that $N^q_{mut} = Lq_{low} + z_\alpha\sqrt{\mathrm{Var}(N^{q_{low}}_{mut})}$ and a unique q_{high} such that $N^q_{mut} = Lq_{\mathrm{high}} - z_\alpha\sqrt{\mathrm{Var}(N^{q_{high}}_{mut})}$, and*

$$\Pr[q \in [q_{low}, q_{high}]] = 1 - \alpha + \epsilon,$$

where $|\varepsilon| \leq c/L^{1/4}$ and c is a constant that depends only on r_1 and k. In particular, for fixed r_1 and k, we have $\lim_{L\to\infty}(1-\alpha+\epsilon) = 1-\alpha$.

Proof. Given the result in corollary 4, we need only show that q_{low} and q_{high} are well-defined. As such, it is sufficient to show that $Lq+z_\alpha\sqrt{\mathrm{Var}(N^q_{\mathrm{mut}})}$ and $Lq-z_\alpha\sqrt{\mathrm{Var}(N^q_{\mathrm{mut}})}$ are strictly monotonic in q for sufficiently large L. Equivalently, since $q = 1-(1-r_1)^k$, these must be strictly monotonic in r_1 which we consider here. For simplicity, we will write r instead of r_1 and N_{mut} instead of N^q_{mut}. Focusing then on $L(1-(1-r)^k) + z_\alpha\sqrt{\mathrm{Var}(N_{\mathrm{mut}})}$, consider

$$\begin{aligned}
&\frac{\partial}{\partial r}\Big(L(1-(1-r)^k) + z_\alpha\sqrt{\mathrm{Var}(N_{\mathrm{mut}})}\Big) \qquad (3)\\
&= Lk(1-r)^{k-1} + z_\alpha\Big((1-q)\Big(4q + r\Big(L\Big(kr^2 - 2(k-1)r - 2\Big) + 2k + 2r - 2kr - 6 + 2(1-q)\\
&\cdot\Big(L\Big(2k^2r^2 + kr(2-r) - r + 1\Big) - k^3r^2 + k^2\Big(r^2 - 2r\Big) - 3k(1-r) - r + 3\Big)\Big)\Big)\Big)\\
&\cdot\Big(2r^2(1-r)\Big((1-q)^2\Big(L\Big(r^2 - 2r - 2kr^2\Big) + k^2r^2 + kr(2-r) + 2(1-r)\Big)\\
&+ (1-q)(Lr(2-r) - 2(1-r))\Big)^{1/2}\Big)^{-1},
\end{aligned}$$

After a (tedious) series expansion of the right side of the equality in eq. (3) about $L = \infty$, we find that $\frac{\partial}{\partial r}\Big(L(1-(1-r)^k) + z_\alpha\sqrt{\mathrm{Var}(N_{\mathrm{mut}})}\Big) = kL(1-r)^{k-1} + o(L)$. As such, $L(1-(1-r)^k) + z_\alpha\sqrt{\mathrm{Var}(N_{\mathrm{mut}})}$ is increasing as a function of r as $L \to \infty$. The case of showing that $L(1-(1-r_1)^k) - z_\alpha\sqrt{\mathrm{Var}(N_{\mathrm{mut}})}$ is also increasing proceeds in an entirely analogous fashion. □

Theorem 6. *Consider the sketching simple mutation model with known parameters s, k, $L \geq k$, r_1, and output $\hat{J}$. Let $0 < \alpha < 1$ and let $m \geq 2$ be an integer. For $0 \leq i \leq m$, let $n^i_l = Lq - z_{i/m}\sqrt{\mathrm{Var}(N_{mut})}$ and $n^i_h = Lq + z_{i/m}\sqrt{\mathrm{Var}(N_{mut})}$. Let*

$$j_{high} = s^{-1}\min\left\{a \geq 0 : m\alpha > \sum_{i:n^i_l>0} F_{\lceil n^i_l\rceil}(a) + \sum_{i:n^i_h\leq L} F_{\lceil n^{i-1}_h\rceil}(a)\right\};\ \textit{and}$$

$$j_{low} = s^{-1}\max\left\{a \leq s : m(2-\alpha) < \sum_{i:n^i_l>0} F_{\lfloor n^{i-1}_l\rfloor}(a) + \sum_{i:n^i_h\leq L} F_{\lfloor n^i_h\rfloor}(a)\right\}.$$

Then, assuming that r_1 and k are independent of L, and $m = o(L^{1/4})$,

$$\lim_{L\to\infty}\Pr[j_{low} \leq \hat{J} \leq j_{high}] = 1-\alpha.$$

Proof. Recall the definition of A and B from the definition of the minhash Jaccard estimator. First, we argue that an element of $(A\cup B)_S$ is in $A_S \cap B_S$ iff it corresponds to a non-mutated k-span (i.e. iff $x = shared_i$ for some i). Consider an element $x \in (A\cup B)_S$. If x corresponds to a mutated k-span (i.e. $x = a\text{-}distinct_i$ or $x = b\text{-}distinct_i$ for some i), then $x \notin A\cap B$ and so $x \notin A_S \cap B_S$.

If x does not correspond to a mutated k-span (i.e. $x = shared_i$ for some i), then $x \in A \cap B$ and $x \in A_S \cap B_S$ as well.

Next, let J' be the random variable corresponding to $|(A \cup B)_S \cap A_S \cap B_S|$. The minhash Jaccard estimator can then be expressed as $\hat{J} = J'/s$. Note that J' contains randomness due to both the mutation process and to the choice of the minhash permutation. For ease of notation we set $N = N_{\text{mut}}$. We claim that the distribution of J', conditioned on $N = n$, is hypergeometric $H(L+n, L-n, s)$. To see this, recall from the discussion in the previous paragraphs that an element of $(A \cup B)_S$ is in $A_S \cap B_S$ only when it corresponds to a non-mutated k-span, and, on the event that $N = n$, there are exactly $L - n$ non-mutated k-spans and a total of $L + n$ k-spans to be hashed. We assume the hash values are assigned with a random permutation. Equivalently, we can generate the hash values by repeatedly assigning the smallest available hash value to an element chosen uniformly at random among those that have not been hashed yet. Then, from the first s elements, the probability that exactly a of those are selected from the set of the $L-n$ non-mutated k-spans is:

$$\Pr[J' = a \mid N = n] = \frac{\binom{L-n}{a}\binom{2n}{s-a}}{\binom{L+n}{s}},$$

which corresponds to the hypergeometric $H(L+n; L-n, s)$ probability function.

Our goal is to deduce a confidence interval for $\hat{J}$, or equivalently for J'. This can be easily done if we could compute:

$$B(a) \triangleq \Pr[J' \geq a] = \sum_{n=0}^{L} F_n(a) \Pr[N = n].$$

However, since we do not have an expression for $\Pr[N = n]$, we will instead obtain an upper and lower bound on $B(a)$ using Lemma 3. For $1 \leq i \leq m$, let $p_l^i = \Pr[N \in [n_l^{i-1}, n_l^i)]$ and let $p_h^i = \Pr[N \in [n_h^i, n_h^{i-1})]$. Note that $p_l^i = 0$ if $n_l^i \leq 0$ and p_h^i if $n_h^i > L$. Using the law of total probability, we can write

$$\begin{aligned}
B(a) &= \sum_{n=0}^{L} \sum_{i=1}^{m} \Big(F_n(a) p_l^i \Pr[N = n \mid N \in [n_l^{i-1}, n_l^i)] + F_n(a) p_h^i \Pr[N = n \mid N \in [n_h^i, n_h^{i-1})] \Big) \\
&= \sum_{i=1}^{m} \sum_{n=0}^{L} \Big(F_n(a) p_l^i \Pr[N = n \mid N \in [n_l^{i-1}, n_l^i)] + F_n(a) p_h^i \Pr[N = n \mid N \in [n_h^i, n_h^{i-1})] \Big) \\
&= \sum_{i=1}^{m} p_l^i \sum_{n=0}^{L} F_n(a) \Pr[N = n \mid N \in [n_l^{i-1}, n_l^i)] + \sum_{i=1}^{m} p_h^i \sum_{n=0}^{L} F_n(a) \Pr[N = n \mid N \in [n_h^i, n_h^{i-1})]
\end{aligned}$$

Observe that $F_n(a)$ is a non-increasing function with respect to n; this is because increasing n in $H(L+n; L-n, s)$ has the overall effect of reducing the probability of success, since the population size is increased and the number of successes is decreased. Using this, we can find upper and lower bounds for $B(a)$ as follows.

$$\begin{aligned}
B(a) \leq \sum_{i: n_l^i > 0} p_l^i F_{\lfloor n_l^{i-1} \rfloor}(a) \sum_{n=0}^{L} \Pr[N = n \mid N \in [n_l^{i-1}, n_l^i)] + \\
\sum_{i: n_h^i \leq L} p_h^i F_{\lfloor n_h^i \rfloor}(a) \sum_{n=0}^{L} \Pr[N = n \mid N \in [n_h^i, n_h^{i-1})]
\end{aligned}$$

$$\leq \sum_{i:n_l^i>0} p_l^i F_{\lfloor n_l^{i-1} \rfloor}(a) + \sum_{i:n_h^i \leq L} p_h^{i-1} F_{\lfloor n_h^i \rfloor}(a).$$

Similarly, we obtain $B(a) \geq \sum_{i:n_l^i>0} p_l^i F_{\lceil n_l^i \rceil}(a) + \sum_{i:n_h^i \leq L} p_h^i F_{\lceil n_h^{i-1} \rceil}(a)$.

We now approximate p_h^i and p_l^i as follows. Observe that when $p_h^i > 0$, we have $p_h^i = \Pr[N \in [n_h^i, n_h^{i-1})] = \Pr[N \geq n_h^i] - \Pr[N \geq n_h^{i-1}]$. By Corollary 4, $\Pr[N \geq n_h^i] = i/(2m) - \varepsilon_1/2$ and $\Pr[N \geq n_h^{i-1}] = (i-1)/(2m) - \varepsilon_2/2$; hence, $p_h^i = 1/(2m) - (\varepsilon_1 - \varepsilon_2)/2$. Here, ε_1 and ε_2 are constants whose absolute value is bounded by $\varepsilon_{\max} = c/L^{1/4}$, with $c > 0$ a constant that depends only on q and k. Hence, $p_h^i \in 1/(2m) \pm \varepsilon_{\max}$. Analogously, we have $p_l^i \in 1/(2m) \pm \varepsilon_{\max}$.

This allows us to further simplify the bounds for $B(a)$:

$$\begin{aligned} B(a) &\leq \sum_{i:n_l^i>0} p_l^i F_{\lfloor n_l^{i-1} \rfloor}(a) + \sum_{i:n_h^i \leq L} p_h^{i-1} F_{\lfloor n_h^i \rfloor}(a) \\ &\leq \left(\frac{1}{2m} + \varepsilon_{\max}\right) \left(\sum_{i:n_l^i>0} F_{\lfloor n_l^{i-1} \rfloor}(a) + \sum_{i:n_h^i \leq L} F_{\lfloor n_h^i \rfloor}(a) \right) \triangleq B_h(a), \end{aligned}$$

and

$$\begin{aligned} B(a) &\geq \sum_{i=1}^{m} \left(p_l^i F_{\lceil n_l^i \rceil}(a) + p_h^{i-1} F_{\lceil n_h^{i-1} \rceil}(a) \right) \\ &\geq \left(\frac{1}{2m} - \varepsilon_{\max}\right) \left(\sum_{i:n_l^i>0} F_{\lceil n_l^i \rceil}(a) + \sum_{i:n_h^i \leq L} F_{\lceil n_h^{i-1} \rceil}(a) \right) \triangleq B_l(a). \end{aligned}$$

Let $a_{\text{MAX}} = \min\{a \geq 0 : \alpha/2 > B_l(a)\}$ and $a_{\text{MIN}} = \max\{a \leq s : \alpha/2 > 1 - B_h(a)\}$. Then, $\Pr[J' \in [a_{\text{MIN}}, a_{\text{MAX}}]] = 1 - \alpha$ and so $\Pr[\hat{J} \in [a_{\text{MIN}}/s, a_{\text{MAX}}/s]] = 1 - \alpha$. The theorem then follows by observing that when r_1, k are independent of L and $m = o(L^{1/4})$, we have $\lim_{L\to\infty} \varepsilon_{\max} m = 0$ and so $a_{\min}/s \to j_{\text{low}}$, and $a_{\max}/s \to j_{\text{high}}$. □

Theorem 7. *For fixed k, r_1, α, m, and a given observed value of $\hat{J}$, there exists an L large enough such that there exist unique intervals $[q_{low}^-, q_{low}^+]$ and $[q_{high}^-, q_{high}^+]$ such that $q_{high}^+ \geq q_{low}^-$, $j_{low}(\hat{q}) = \hat{J}$ if and only if $\hat{q} \in [q_{low}^-, q_{low}^+]$, and $j_{high}(\hat{q}) = \hat{J}$ if and only if $\hat{q} \in [q_{high}^-, q_{high}^+]$. Moreover, assuming that r_1, k and m are independent of L, we have*

$$\lim_{L\to\infty} \Pr[q \in [q_{low}^-, q_{high}^+]] = 1 - \alpha.$$

Proof. Recall from Theorem 6 that $n_l^i = Lq - z_{i/m}\sqrt{\text{Var}(N_{\text{mut}})}$ and $n_h^i = Lq + z_{i/m}\sqrt{\text{Var}(N_{\text{mut}})}$. First, observe from Theorem 2 that $\text{Var}(N_{\text{mut}}) = cL + o(L)$, where c is a constant depending only on k and r_1. Consequently, $\sqrt{\text{Var}(N_{\text{mut}})} = o(L)$ and so, for fixed k, r_1, α, and m, there exists an L sufficiently large such that, for all $i = 0, \ldots, m$, $n_h^i \in [0, L]$ and $n_l^i \in [0, L]$. Therefore, for all values of q, there exists an L sufficiently large such that the summations in the definition of j_{low} and j_{high} are over $0 \leq i \leq m$. Second, in the proof of Theorem 5 we established that n_l^i and n_h^i are increasing with q provided L is sufficiently large. Therefore, the parameters in the subscripts of the F terms of j_{low} and j_{high} are also increasing with q, when L is sufficiently large. Third, observe that $F_n(a)$ is a non-increasing function of n and of a. The fact that it is a non-increasing function of n we already observed in the proof of Theorem 6. The fact that it is a non-increasing

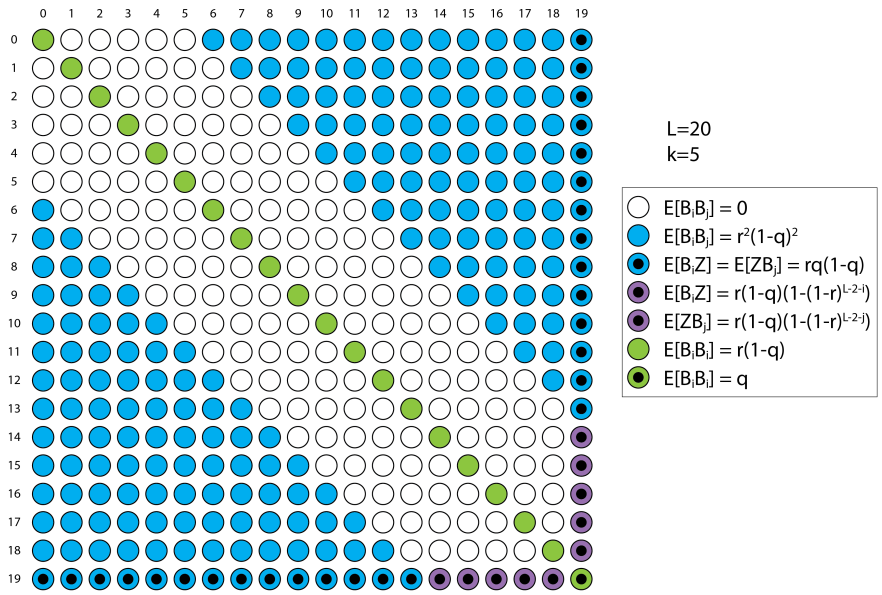

Figure S1: Illustration of the joint probabilities of X_i and X_j, i.e. the terms of the sum in the derivation of $\mathrm{Var}(N_{\mathrm{isl}})$ in Theorem 9. In this example, $L = 20$ and $k = 5$.

function of a follows trivially from its definition. Combining these three observations, we deduce that j_{low} are j_{high} non-increasing functions of q. Therefore, they take on a certain value (i.e. $\hat{J}$) for a unique range of the domain, implying the first assertion of the theorem. The second assertion of the theorem then follows from Theorem 6. □

Theorem 9. *For $L \geq k+3$,* $\mathrm{Var}(N_{isl}) = Lr_1(1-q)(1-r_1(1-q)(2k+1)) + k^2 r_1^2(1-q)^2 + k(r_1(3r_1+2)(1-q)^2) + (1-q)((1-q)r_1^2 - q - r_1)$.

Proof. For convenience, we will define a random variable X_i for $0 \leq i \leq L-1$ and let $X_i = B_i$ for $i < L-1$ and $X_{L-1} = Z$. Also, for notational simplicity, write r for r_1. Figure S1 visualizes the joint probabilities of all X_i's, as given in Lemma 8. Using the figure as a guide, we proceed with the derivation.

$$
\begin{aligned}
\mathrm{Var}[N_{\mathrm{island}}] &= E[N_{\mathrm{island}}^2] - E[N_{\mathrm{island}}]^2 \\
&= \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} E[X_i X_j] - (r(1-q)(L-1)+q)^2 \\
&= \sum_{i=0}^{L-2} E[X_i] \\
&\quad + E[X_{L-1}] \\
&\quad + 2\sum_{i=0}^{L-3-k}\sum_{j=i+k+1}^{L-2} E[X_i X_j] \\
&\quad + 2\sum_{i=0}^{L-2-k} E[X_i X_{L-1}]
\end{aligned}
$$

$$
\begin{aligned}
&\quad + 2 \sum_{i=L-1-k}^{L-2} E[X_i X_{L-1}] \\
&\quad - (r(1-q)(L-1)+q)^2 \\
&= \sum_{i=0}^{L-2} r(1-q) \\
&\quad + q \\
&\quad + 2 \sum_{i=0}^{L-3-k} \sum_{j=i+k+1}^{L-2} r^2(1-q)^2 \\
&\quad + 2 \sum_{i=0}^{L-2-k} rq(1-q) \\
&\quad + 2 \sum_{i=L-1-k}^{L-2} r(1-q)(1-(1-r)^{L-2-i}) \\
&\quad - (r(1-q)(L-1)+q)^2 \\
&= \sum_{i=0}^{L-2} r(1-q) \\
&\quad + q \\
&\quad + 2 \sum_{i=0}^{L-3-k} \sum_{j=i+k+1}^{L-2} r^2(1-q)^2 \\
&\quad + 2 \sum_{i=0}^{L-2-k} rq(1-q) \\
&\quad + 2 \sum_{i=0}^{k-1} r(1-q)(1-(1-r)^{i}) \\
&\quad - (r(1-q)(L-1)+q)^2 \\
&= \sum_{i=0}^{L-2} r(1-q) \\
&\quad + q \\
&\quad + 2 \sum_{i=0}^{L-3-k} \sum_{j=i+k+1}^{L-2} r^2(1-q)^2 \\
&\quad + 2 \sum_{i=0}^{L-2-k} rq(1-q) \\
&\quad + 2 \sum_{i=0}^{k-1} r(1-q) - 2 \sum_{i=0}^{k-1} r(1-q)(1-r)^{i} \\
&\quad - (r(1-q)(L-1)+q)^2 \\
&= (L-1)r(1-q)
\end{aligned}
$$

$$
\begin{aligned}
&\quad + q \\
&\quad + 2\frac{(L-k-1)(L-k-2)}{2} r^2(1-q)^2 \\
&\quad + 2(L-1-k)rq(1-q) \\
&\quad + 2kr(1-q) - 2r(1-q)\frac{1-(1-r)^k}{1-(1-r)} \\
&\quad - (r^2(1-q)^2(L-1)^2 + 2rq(1-q)(L-1) + q^2) \\
&= (L-1)r(1-q) \\
&\quad + q \\
&\quad + (L-k-1)(L-k-2)r^2(1-q)^2 \\
&\quad + 2(L-1-k)rq(1-q) \\
&\quad + 2kr(1-q) - 2q(1-q) \\
&\quad - r^2(1-q)^2(L-1)^2 - 2rq(1-q)(L-1) - q^2 \\
&= (L-1)r(1-q) \\
&\quad + q \\
&\quad + (L^2 - (2k+3)L + k^2 + 3k + 2)r^2(1-q)^2 \\
&\quad + 2(L-1-k)rq(1-q) \\
&\quad + 2kr(1-q) - 2q(1-q) \\
&\quad - (L^2 - 2L + 1)r^2(1-q)^2 - 2rq(1-q)(L-1) - q^2 \\
&= Lr(1-q) - r(1-q) \\
&\quad + q \\
&\quad + L^2r^2(1-q)^2 - (2k+3)Lr^2(1-q)^2 + k^2r^2(1-q)^2 + 3kr^2(1-q)^2 + 2r^2(1-q)^2 \\
&\quad + 2Lrq(1-q) - 2rq(1-q) - 2krq(1-q) \\
&\quad + 2kr(1-q) - 2q(1-q) \\
&\quad - L^2r^2(1-q)^2 + 2Lr^2(1-q)^2 - r^2(1-q)^2 - 2Lrq(1-q) + 2rq(1-q) - q^2 \\
&= L^2(r^2(1-q)^2 - r^2(1-q)^2) \\
&\quad + L(r(1-q) - (2k+3)r^2(1-q)^2 + 2rq(1-q) - 2rq(1-q) + 2r^2(1-q)^2) \\
&\quad + k^2r^2(1-q)^2 \\
&\quad + k(3r^2(1-q)^2 - 2rq(1-q) + 2r(1-q)) \\
&\quad + (2-1)r^2(1-q)^2 \\
&\quad + (2-2)rq(1-q) \\
&\quad - 2q(1-q) \\
&\quad - r(1-q) \\
&\quad - q^2 \\
&\quad + q \\
&= Lr(1-q)(1-(2k+1)r(1-q)) \\
&\quad + k^2r^2(1-q)^2 \\
&\quad + kr(3r+2)(1-q)^2
\end{aligned}
$$

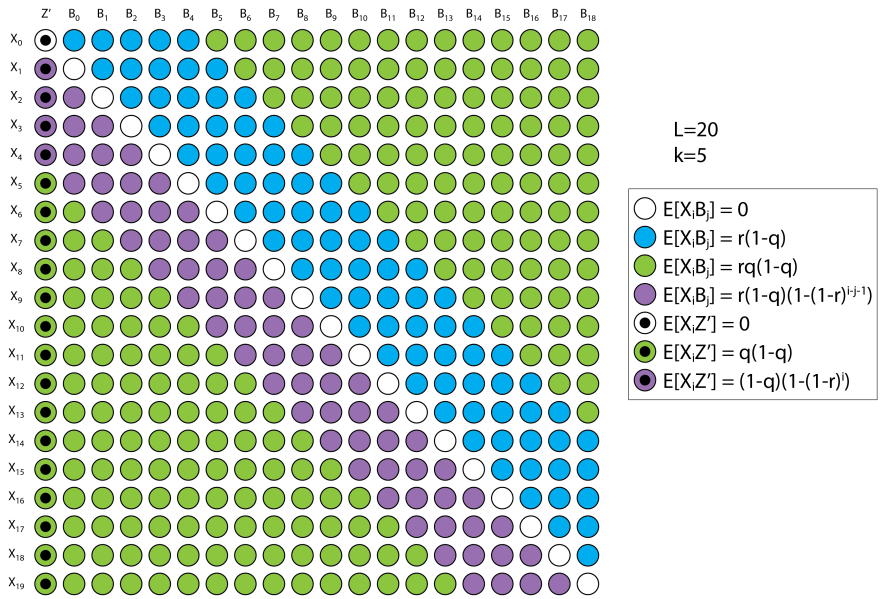

Figure S2: Illustration of the joint probabilities of X_i, B_j, and Z', i.e. the terms of the sum in the derivation of $\mathrm{Cov}\,(N_{\mathrm{ocean}}, N_{\mathrm{mut}})$ in Theorem 11. In this example, $L = 20$ and $k = 5$.

$$+ (1-q)((1-q)r^2 - q - r).$$

□

Theorem 12. $\mathrm{E}[N_{ocean}] = Lr_1(1-q) + (1-q)(1-r_1)$ *and, for* $L \geq k+3$,

$$\begin{aligned}\mathrm{Var}(N_{ocean}) = {} & L(1-q)(r_1 - r_1^2(1-q)^2(2k+1)) + k^2 r_1^2 (1-q)^2 \\ & - k(1-q)^2(r_1(2 + r_1(2Lq-3))) + (1-q)(q - r_1 - r_1^2(1-q)(Lq-1)).\end{aligned}$$

Proof. Let us define Z' as an indicator for the event that the first k-span (K_0) is not mutated. Hence $\mathrm{E}[Z'] = (1-r_1)^k = (1-q)$. Observe that every ocean begins either the start of the interval or a right border. Therefore, the the number of oceans is $N_{\mathrm{ocean}} = Z' + \sum_{i=0}^{L-2} B_i$. Thus $\mathrm{E}[N_{\mathrm{ocean}}] = (1-q) + (L-1)r_1(1-q) = (1-r_1)(1-q) + Lr_1(1-q)$. For the variance, the derivation is equivalent to replacing Z with Z' in the derivation of Theorem 9 and we therefore omit the proof here. □

Theorem 11. $\mathrm{E}[C_{ber}] = \frac{1-q}{L+k-1}\left(L(1+r_1(k-1)) + (1-r_1)(k-1)\right)$ *and, for* $L \geq k+3$, $\mathrm{Var}(C_{ber}) = \frac{(1-q)(cL+d)}{r^2(L+k-1)}$, *where*

$$\begin{aligned}c &= 2rq + r^2(-3q - 2k + 4kq) + r^3(k-1)(4kq - 3k - 1) + r^4(1-q)(k-1)^2(-2k-1); \text{ and} \\ d &= -2q + 2r(q + k - kq) + r^2(k-1)(k-q) \\ &\quad + r^3(k-1)(3k - 4kq + 1) + r^4(k-1)^2(1-q)(k^2 + 3k + 1).\end{aligned}$$

Proof. Throughout, we write r instead of r_1 for simplicity. Recall that $C_{\mathrm{ber}} = (L - N_{\mathrm{mut}} + (k-1)N_{\mathrm{ocean}})/(L+k-1)$. Applying linearity of expectation together with eq. (1) and Theorem 12,

$$\mathrm{E}[C_{\mathrm{ber}}] = \frac{L - \mathrm{E}[N_{\mathrm{mut}}] + (k-1)\mathrm{E}[N_{\mathrm{ocean}}]}{L+k-1}$$

$$= \frac{L - Lq + (k-1)(Lr(1-q) + (1-q)(1-r))}{L+k-1}$$
$$= \frac{L}{L+k-1}(1-q)(1+r(k-1)) + \frac{(1-q)(1-r)(k-1)}{L+k-1}$$

Applying the distributive properties of variance to the definition of C_{ber} we get:

$$\begin{aligned}
\text{Var}(C_{\text{ber}}) &= \text{Var}((L - N_{\text{mut}} + (k-1)N_{\text{ocean}})/(L+k-1)) \\
&= (L+k-1)^{-2}\,\text{Var}((k-1)N_{\text{ocean}} - N_{\text{mut}})) \\
&= (L+k-1)^{-2}\left((k-1)^2\,\text{Var}(N_{\text{ocean}}) + \text{Var}(N_{\text{mut}}) - 2(k-1)\,\text{Cov}(N_{\text{ocean}}, N_{\text{mut}})\right).
\end{aligned}$$

We only need to compute the covariance. We will use the same definition of variables as previously in Sections 3 and 6. In particular, X_i is a random variable indicating that K_i was mutated, Z' indicates that K_0 has not mutated, and B_i indicates that K_i was mutated and K_{i+1} has not. Note that $Z' = 1 - X_0$. Figure S2 visualizes the joint probabilities of all X_i's, B_j's, and Z. We then compute

$$\begin{aligned}
\text{Cov}\,(N_{\text{ocean}}, N_{\text{mut}}) &= \text{E}[N_{\text{mut}}N_{\text{ocean}}] - \text{E}[N_{\text{mut}}]\text{E}[N_{\text{ocean}}] \\
&= \text{E}\left[\sum_{i=0}^{L-1} X_i \left(Z' + \sum_{j=0}^{L-2} B_j\right)\right] - Lq(Lr_1(1-q) + (1-q)(1-r_1)) \\
&= \text{E}\left[Z'\sum_{i=0}^{L-1} X_i\right] + \text{E}\left[\sum_{i=0}^{L-1} X_i \sum_{j=0}^{L-2} B_j\right] - Lq(Lr_1(1-q) + (1-q)(1-r_1)). \quad (4)
\end{aligned}$$

Observe that when $i = 0$, $\text{E}[Z'X_i] = 0$. When $1 \le i \le k-1$, $\text{E}[Z'X_i] = (1-q)(1-(1-r)^i)$ since the left-most i nucleotides are not mutated when $Z' = 1$. When $k \le i \le L-1$, Z' and X_i are independent so $\text{E}[Z'X_i] = q(1-q)$. Thus, calculating the first sum in equation (4), we obtain

$$\begin{aligned}
\text{E}\left[\sum_{i=0}^{L-1} Z'X_i\right] &= \sum_{i=k}^{L-1} q(1-q) + \sum_{i=1}^{k-1}(1-q)\left(1-(1-r)^i\right) \\
&= (L-k)\left(1-(1-r)^k\right)(1-r)^k + \frac{\left((1-r)^k + kr - 1\right)(1-r)^k}{r}
\end{aligned}$$

For the second sum in equation (4), observe that $B_j = 1$ implies that $X_j = 1$ and $X_{j+1} = 0$. Furthermore, for $2 \le d \le k$, if $X_{j+d} = 1$, then the leftmost $k+1+d$ nucleotides are unmutated, so from the law of total probability, $\text{E}[B_jX_{j+d}] = (1-(1-r)^{d-1})r(1-q)$. Lastly, if $B_j = 1$, then for all integers i such that $\max\{0, j-k+1\} \le i \le j$, $X_i = 1$ as well due to the mutation at position j. Hence $\text{E}[B_jX_i] = \text{E}[B_j] = r(1-q)$. Using these observations, we obtain

$$\begin{aligned}
\text{E}\left[\sum_{i=0}^{L-1} X_i \sum_{j=0}^{L-2} B_j\right] &= \sum_{i=0}^{L-1}\sum_{j=0}^{L-2} \text{E}\left[X_iB_j\right] \\
&= \sum_{i=0}^{L-k-2}\sum_{j=i+k}^{L-2} q(1-q)r \\
&+ \sum_{i=0}^{L-k-1}\sum_{j=i}^{i+k-1}(1-q)r + \sum_{i=L-k}^{L-2}\sum_{j=i}^{L-2}(1-q)r
\end{aligned}$$

$$
\begin{aligned}
&+ \sum_{j=0}^{L-k-1} \sum_{i=j+2}^{j+k} (1-q)r\left(1-(1-r)^{i-j-1}\right) \\
&+ \sum_{j=L-k}^{L-3} \sum_{i=j+2}^{L-1} (1-q)r\left(1-(1-r)^{i-j-1}\right) \\
&+ \sum_{j=0}^{-k+L-2} \sum_{i=j+k+1}^{L-1} q(1-q)r \\
&= \frac{1}{2}(1-q)qr(k-L)(k-L+1) \\
&+ k(1-q)r(L-k) + \frac{1}{2}(k-1)k(1-q)r \\
&+ \frac{(1-q)\left(k^2r^2 - kr^2 - 2(1-q) - 2kr + 2\right)}{2r} \\
&+ (1-q)(L-k)\left((1-q) + kr - 1\right) \\
&+ \frac{1}{2}(1-q)qr(k-L)(k-L+1)
\end{aligned}
$$

Putting all of this together and simplifying, we obtain

$$
\begin{aligned}
&\mathrm{Var}(C_{\mathrm{ber}}) = \\
&r^{-2}(L+k-1)^{-2}(1-r)^k\Big(r^2\Big(k^2 + (-4kL+k+3L-1)(1-r)^k + 2kL - 2k - 3L + 1\Big) \\
&+ (k-1)^2r^4(k(k-2L+3) - L + 1)(1-r)^k - (k-1)(L-1)r^3\Big(k\Big(4(1-r)^k - 1\Big) \\
&+ 1\Big) + 2r\Big((k-L-1)(1-r)^k + L + 1\Big) + 2\Big((1-r)^k - 1\Big)\Big).
\end{aligned}
$$

Factoring out the L terms in the numerator, we get

$$
\begin{aligned}
&\mathrm{Var}(C_{\mathrm{ber}}) = \frac{1-q}{r^2(L+k-1)} \cdot \\
&\cdot \left(L(2rq + r^2(-3q - 2k + 4kq) + r^3(k-1)(4kq - 3k - 1) + r^4(1-q)(k-1)^2(-2k-1))\right. \\
&- 2q + 2r(q + k - kq) + r^2(k-1)(k-q) \\
&+ r^3(k-1)(3k - 4kq + 1) + r^4(k-1)^2(1-q)(k^2 + 3k + 1))
\end{aligned}
$$

□

A.2 Experimental results: extra tables and figures

r_1	k	L	$\mathrm{E}[N_{\mathrm{mut}}]$	$\overline{N_t} = t^{-1}\sum_{i=1}^{t} N^i$	$\mathrm{Var}(N_{\mathrm{mut}})$	$s_t^2 = t^{-1}\sum_{i=1}^{t}(N^i - \overline{N_t})^2$
0.001	21	100	2.1	2.1	40	39
0.001	21	1,000	20.8	20.7	423	419
0.001	21	10,000	207.9	207.8	4,257	4,304
0.001	51	100	5.0	4.9	199	199
0.001	51	1,000	49.7	49.8	2,350	2,417
0.001	51	10,000	497.5	498.9	23,863	24,072
0.001	100	100	9.5	9.3	567	557
0.001	100	1,000	95.2	95.9	8,191	8,116
0.001	100	10,000	952.1	957.3	84,427	83,041
0.010	21	100	19.0	19.0	291	287
0.010	21	1,000	190.3	190.5	3,101	3,118
0.010	21	10,000	1,902.7	1,905.2	31,198	31,225
0.010	51	100	40.1	39.9	939	939
0.010	51	1,000	401.0	400.9	11,027	11,014
0.010	51	10,000	4,010.4	4,013.1	111,908	113,098
0.010	100	100	63.4	63.4	1,350	1,362
0.010	100	1,000	634.0	634.1	18,794	19,183
0.010	100	10,000	6,339.7	6,345.9	193,237	188,818
0.100	21	100	89.1	89.0	127	130
0.100	21	1,000	890.6	890.4	1,341	1,373
0.100	21	10,000	8,905.8	8,907.5	13,479	13,589
0.100	51	100	99.5	99.5	8	8
0.100	51	1,000	995.4	995.5	85	84
0.100	51	10,000	9,953.6	9,954.3	855	830
0.200	21	100	99.1	99.1	8	7
0.200	21	1,000	990.8	990.9	78	78
0.200	21	10,000	9,907.8	9,908.0	786	791

Table S1: Validation of Equation (1) and Theorem 2, using $t = 10,000$ trials. For each row, we show the value of $\mathrm{E}[N_{\mathrm{mut}}]$ given by Equation (1), the sample average of N_{mut} over all trials ($\overline{N_t}$), the value of $\mathrm{Var}[N_{\mathrm{mut}}]$ given by Theorem 2, and the sample variance of all the trials (s_t^2). Here, N^i is observed N_{mut} for the ith trial.

	$L = 100$				$L = 1,000$				$L = 10,000$			
$r_1 =$	0.001	0.01	0.1	0.2	0.001	0.01	0.1	0.2	0.001	0.01	0.1	0.2
$k = 100$	0.94	1.00	NA	NA	0.99	0.99	NA	NA	0.99	0.99	NA	NA
$k = 51$	0.93	1.00	0.98	NA	0.99	0.99	0.96	NA	0.99	0.99	0.98	NA
$k = 21$	0.92	0.99	0.98	0.97	0.98	0.99	0.99	0.98	0.99	0.99	0.99	0.99

Table S2: The accuracy of the confidence intervals for r_1 predicted by Corollary 4, for $\alpha = 0.01$ and for various values of L, r_1, and k. NA indicates the experiment was not run because the parameters were not of interest (precisely, $\lceil \mathrm{E}[N_{\mathrm{mut}}] \rceil = L$). The number of replicates was 10,000 for all experiments.

	$L = 100$				$L = 1,000$				$L = 10,000$			
$r_1 =$	0.001	0.01	0.1	0.2	0.001	0.01	0.1	0.2	0.001	0.01	0.1	0.2
$k = 100$	0.90	0.86	NA	NA	0.94	0.90	NA	NA	0.91	0.90	NA	NA
$k = 51$	0.91	0.94	0.97	NA	0.93	0.90	0.93	NA	0.91	0.90	0.94	NA
$k = 21$	0.91	0.94	0.92	0.94	0.93	0.90	0.90	0.93	0.92	0.90	0.90	0.90

Table S3: The accuracy of the confidence intervals for r_1 predicted by Corollary 4, for $\alpha = 0.10$ and for various values of L, r_1, and k. NA indicates the experiment was not run because the parameters were not of interest (precisely, $\lceil \mathrm{E}[N_{\mathrm{mut}}] \rceil = L$). The number of replicates was 10,000 for all experiments.

Sketch size	$r_1 = .05$, $q = .659$	$r_1 = .15$, $q = .967$	$r_1 = .25$, $q = .998$
100	0.97	1.00	1.00
1,000	0.97	0.96	1.00
10,000	0.96	0.96	0.97
100,000	0.94	0.95	0.96

Table S4: The accuracy of confidence intervals predicted by Theorem 6 on a real *E.coli* sequence. For each sketch size and r_1 value, we show the number of trials for which the true r_1 falls within the predicted confidence interval. Here, $\alpha = 0.05$, $k = 21$, and the sketch size s and r_1 are varied as shown. The number of trials for each cell is 1,000, and $m = 100$ for Theorem 6. *E.coli* strain K-12 substr. MG1655 was used.

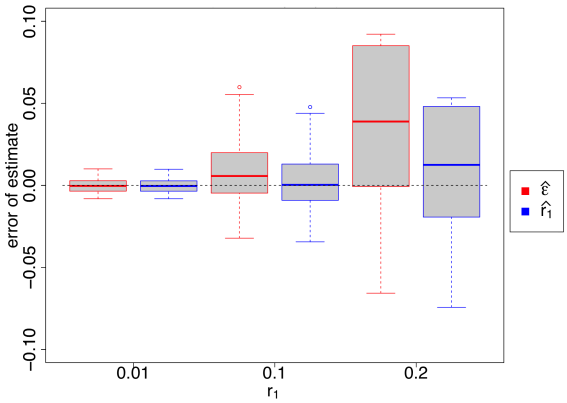

Figure S3: Estimates of sequence divergence as done by mimimap2 ($\hat{\epsilon}$) and by our approach ($\hat{r}_1$). This is similar to Figure 2 but with sequence lengths of 1kbp instead of 10kbp.