Assembly-free and alignment-free sample identification using genome skims

Shahab Sarmashghi[1], Kristine Bohmann[2,3], M. Thomas P. Gilbert[2,4], Vineet Bafna[5], and Siavash Mirarab[1]

[1]Department of Electrical & Computer Engineering University of California, San Diego, La Jolla, CA 92093, USA
[2]Evolutionary Genomics, Natural History Museum of Denmark, University of Copenhagen, Copenhagen, Denmark
[3]School of Biological Sciences, University of East Anglia, Norwich, Norfolk, UK
[4]Norwegian University of Science and Technology, University Museum, 7491 Trondheim, Norway
[5]Department of Computer Science & Engineering, University of California, San Diego, La Jolla, CA 92093, USA

Abstract

The ability to quickly and inexpensively describe the taxonomic diversity in an environment is critical in this era of rapid climate and biodiversity changes. The currently preferred molecular technique is (meta)barcoding in which taxonomically informative plasmid/mitochondrial markers are sequenced. It is low-cost, and widely used, but has drawbacks. As sequencing costs continue to fall, an alternative approach based on *genome-skimming* has been proposed. This approach first applies low-pass (100Mb – several Gb per sample) sequencing to voucher and/or query samples and then recovers marker genes and/or organelle genomes computationally. In contrast, we suggest the use of the unassembled sequence data for taxonomic identification using an alignment-free approach based on the k-mer decomposition of the sequencing reads. Our approach is motivated by earlier work that connects genomic distance to the Jaccard index on k-mer collections, but improves upon prior work through a careful modeling of the impact of low-coverage, sequencing error, and other factors on the Jaccard index. Our tool, Skmer, estimates genomic distance between two organisms represented by their k-mer collections obtained from the genome-skims, and uses distance estimates to match a genome-skim query to a reference collection. Skmer shows excellent performance in our simulation studies, and makes the assembly-free approach to genome-skimming a viable alternative to the traditional barcoding. The Skmer software is made publicly available on `https://github.com/shahab-sarmashghi/Skmer.git`

Keywords. Assembly-free, Alignment-free, DNA Barcoding, DNA Metabarcoding, Genome-skimming, DNA reference databases.

Corresponding Authors:
Siavash Mirarab, smirarab@ucsd.edu
Vineet Bafna, vbafna@cs.ucsd.edu

1 Introduction

The ability to quickly and inexpensively study the taxonomic diversity in an environment is critical in this era of rapid climate and biodiversity changes. The current molecular technique of choice is (meta)barcoding [1–3]. Traditional (meta)barcoding is based on DNA sequencing of taxonomically informative and group-specific marker genes (e.g., mitochondrial COI [1, 4] and 12S/16S [5, 6] for animals, chloroplast genes like matK for plants [7], and ITS [8] for fungi) that are variable enough for taxonomic identification, but have flanking regions that are sufficiently conserved to allow for PCR amplification using universal primers. Barcoding is used for taxonomic identification of single-species samples. In the case of metabarcoding, the goal is to deconstruct the taxonomic composition of a mixed sample consisting of multiple species. Beyond the barcoding application, the barcoding marker genes have also been used to delimitate species [9] and to infer phylogenies [10, 11].

The accuracy of (meta)barcoding depends on the coverage of the reference database and the method used to search queries against it [3]. To satisfy the coverage requirement, reference databases with millions of barcodes have been generated (e.g., the Barcode of Life Data System, BOLD, for the COI marker [12]). Computational methods for finding the closest match in a reference dataset of markers (e.g., TaxI [13]), and for placement of a query into existing marker trees [14–16] have been developed. However, the traditional approach to (meta)barcoding has drawbacks. PCR for marker gene amplification requires relatively high quality DNA and thus cannot be applied to samples in which the DNA is heavily fragmented. Moreover, since barcode markers are relatively short regions, their phylogenetic signal and identification resolution can be limited [17]. For example, 896 of the 4,174 species of the wasp could not be distinguished from other species using COI barcodes [18]. While low costs have kept PCR-based pipelines attractive, decreasing costs of shotgun sequencing have now made it possible to shotgun sequence 1-2Gb of total DNA per reference specimen sample for as low as $80 [19], even after including sample preparation and labor costs. Therefore, researchers have proposed an alternate method for barcoding a sample that uses low-pass sequencing to generate *genome-skims* [19, 20], and subsequently identifies chloroplast or mitochondrial marker genes or assembles the organelle genome. Reconstructing plastid and mtDNA genomes from low-pass shotgun data is doable because non-nuclear DNA tends to be heavily overrepresented in shotgun sequencing; for example, 10.4% of all reads from the Apocynaceae family of flowering plants were from the chloroplast in one genome-skimming study [20]. Large reference databases based on genome-skimming techniques are under construction (e.g., PhyloAlps [21], NorBol [22], and DNAmark [23] projects).

Most current applications of genome-skimming to species identification require organelle genome assembly, a task that requires relatively time-consuming manual curation steps to ensure that assembly errors are avoided [24]. The current approach also discards a vast proportion of the non-target data, which means reducing the signal. Among the existing genome-skimming projects, the DNAmark project has started to consider an alternative approach. Perhaps instead of only relying on organelle markers, we could use the entire set of reads generated in a genome-skim as the identifier of a species. This approach poses an interesting methodological question: can the unassembled data be used to taxonomically profile reference and query samples in a similar manner to conventional barcoding, but using all available genomic information and saving us from the labor-intensive task of mitochondria/plastid genome assembly? In this paper, we introduce a new method to use low coverage genome-skims of both reference and query samples. Our approach aims to use *all* the generated sequence data and to eliminate the need for marker gene assembly. By

avoiding the assembly step, our approach also reduces the amount of data processing needed for expanding the reference database.

We treat genome-skims simply as low-coverage "bags of reads", both for a collection of reference species and for query samples. The problem is to find the reference genome-skim that matches the query; if an exact match is not found, we seek the closest available match. A more advanced problem, not directly addressed here, is placing the query in a phylogeny of reference species. A yet more difficult challenge, also not addressed here, is decomposing a query genome-skim that contains DNA from several different taxa into its constituent species.

Central to solving these problems is the ability to estimate a *distance* between two genome-skims for low and varied coverage using assembly-free and alignment-free approaches. Alignment-free comparison methods [25–27] have been widely studied, including for phylogenetic reconstruction [25, 28–37]. However, these methods typically assume high coverage, enough to cover the most of the genome with at least one read [38]. The required levels of coverage are not economically feasible for building up large databases of reference genome skims or for general processing of query samples. Like many existing alignment-free methods [39, 40], we decompose all reads into fixed length oligomers (denoted *k-mers* with length k) [41], and use existing tools for computing the k-mer frequencies (e.g., JellyFish [42]). Similar to Ondov *et al.*, we compute the hamming distance using k-mers [41]. Recall that the *Jaccard index* J is a similarity measure between any two sets (e.g. k-mer collections) defined as the size of their intersection divided by the size of their union. Ondov *et al.* describe a tool, Mash [41], in which (a) J is estimated efficiently using a hashing procedure; and, (b) J is translated into an estimate of the hamming distance between two genomes, which in turn, relates to the evolutionary distance. Unfortunately, the estimate of J is impacted by coverage, repeats, sequencing error and other factors, and no current approach works well for low coverage datasets. Here, we develop and implement techniques to correct these errors with the aim of enabling the assembly-free approach to genome-skimming. Our tool, Skmer, shows excellent performance in computing distance, identification, and placement of genome-skim queries on to a reference collection. The assembly-free approach to genome-skimming, therefore, should be further explored as a viable alternative to the current approach.

2 Methods

Consider an idealized model where two genomes are the outcome of a random process that copies a genome and introduces mutations at each position with fixed probability d. Moreover, substitutions are the only allowed mutation. In this case, the per-nucleotide hamming distance D between the two genomes is a random variable (r.v.) with expected value d. We would like to estimate d. While this is a simplified model, we will test the method on real pairs of genomes that differ due to complex mutational processes (also, see Appendix B for extensions). We start with known results connecting the Jaccard index and the hamming distance and then show how these results can be generalized to low coverage genome-skims. Throughout, we present our results succinctly and present derivations and more careful justifications in Appendix A

Jaccard index versus genomic distance. The Jaccard index of subsets A_1 and A_2 is defined as

$$J = \frac{|A_1 \cap A_2|}{|A_1 \cup A_2|} = \frac{|A_1 \cap A_2|}{|A_1| + |A_2| - |A_1 \cap A_2|} \,. \tag{1}$$

Let W be the number of shared k-mers between the two genomes. Note that: $J = \frac{W}{2L-W} \Rightarrow \frac{2J}{1+J} = \frac{W}{L}$, where L is the genome length. Assuming random genomes and no repeats, perhaps justifiably [43], the probability that a changed k-mer exists elsewhere in the genome is vanishingly small for sufficiently large k. Thus, we assume a k-mer is in the shared k-mers set only if no mutation falls on it, an event that has probability $(1-d)^k$. Thus, we can model W as a binomial with probability $(1-d)^k$ and L trials. As Ondov *et al.* [41] pointed out, we can estimate

$$D = 1 - \left(\frac{2J}{J+1}\right)^{\frac{1}{k}} \tag{2}$$

and they further approximate D as $\frac{1}{k}\ln\left(\frac{J+1}{2J}\right)$. To be able to estimate large distances, we avoid the unnecessary approximation and use Equation 2 directly. We skim each genome to obtain k-mer sets A_1, A_2 and estimate J using Equation 1, which can be computed efficiently using a hashing technique used by Mash [41]. Note that, however, Equation 2 assumes a high coverage of the genome so that each k-mer is sampled at least once with very high probability. This assumption is violated for genome-skims in consequential ways. As a simple example, suppose the coverage is low enough that a k-mer is sampled with probability 0.5. Then, even for identical genomes, we estimate J as $\frac{1}{3}$, resulting in a distance estimate of $D \approx 0.032$ for $k = 21$.

2.1 Extending to genome-skims with known low coverage

We now show how Equation 2 can be refined to handle genome-skims despite low and uneven coverage, sequencing error, and varying genome-lengths. We assume that coverage is known (but see the next section).

When the genome is not fully covered, three sources of randomness are at work: mutations and sampling of k-mers from each of the two genomes. Each genome of length L is sequenced independently using randomly distributed short reads of length ℓ at coverages c_1 and c_2 to produce two genome-skims. Under the simplifying assumption that genomes are not repetitive, we choose k to be large enough so that each k-mer is unique with high probability. Therefore, the number of distinct k-mers in each genome is $L - k \simeq L$. The probability of covering each k-mer can be approximated as $\eta_i = 1 - e^{-\lambda_i}$ where $\lambda_i = c_i(1 - k/\ell)$. Modeling the sampling of k-mers as independent Bernoulli trials, $|A_i|$ becomes binomially distributed with parameters η_i and L. By independence, $W = |A_1 \cap A_2|$ also becomes binomially distributed with parameters $\eta_1\eta_2(1-d)^k$ and L. Moreover, $U = |A_1 \cup A_2|$ can also be modeled approximately as a Gaussian with mean $(\eta_1 + \eta_2 - \eta_1\eta_2(1-d)^k)L$. Treating η_1 and η_2 as known and dividing $\frac{W}{L}$ by $\frac{U}{L}$ gives us:

$$J = \frac{W}{U} = \frac{\eta_1\eta_2(1-D)^k}{\eta_1 + \eta_2 - \eta_1\eta_2(1-D)^k} ;$$

thus,

$$D = 1 - \left(\frac{(\eta_1 + \eta_2)}{\eta_1\eta_2}\frac{J}{(1+J)}\right)^{\frac{1}{k}} .$$

Sequencing error. Each error reduces the number of shared k-mers and increases the total number of observed k-mers, and thus can also change the Jaccard index. Let ϵ denote the base-miscall rate. For large k and small ϵ, the probability that an erroneous k-mer produces a non-novel k-mer is negligible. The probability that a k-mers is covered by at least one read, without any error, is approximately

$$\eta_i = 1 - e^{-\lambda_i(1-\epsilon)^k} . \tag{3}$$

Adding up the number of error-free and erroneous k-mers, the total number of k-mers observed from both genomes can again be approximately modeled as a Gaussian with mean $\zeta_i L$ for

$$\zeta_i = \eta_i + \lambda_i(1 - (1 - \epsilon)^k) \ . \tag{4}$$

Just as before, we can simply estimate D by solving for it in

$$J = \frac{\eta_1\eta_2(1 - D)^k}{\zeta_1 + \zeta_2 - \eta_1\eta_2(1 - D)^k} \ . \tag{5}$$

When the coverage is sufficiently high, each k-mer will be covered by multiple reads with high probability, and low-abundance k-mers can be safely considered as erroneous. Mash has an option to filter out k-mers with abundances less than some threshold m to remove k-mers that are likely to be erroneous. In this case,

$$\zeta = \eta = 1 - \sum_{t=0}^{m-1} \frac{(\lambda(1 - \epsilon)^k)^t}{t!} e^{-\lambda(1-\epsilon)^k} \tag{6}$$

assuming all erroneous k-mers are removed. For instance, filtering single-copy k-mers (i.e., $m = 2$) gives us:

$$\zeta = \eta = 1 - e^{-\lambda(1-\epsilon)^k} - \lambda(1 - \epsilon)^k e^{-\lambda(1-\epsilon)^k}$$

and the Jaccard index follows the same equation as (5). Since this filtering approach only works for high coverage, we filter low coverage k-mers only when our estimated coverage is higher than a threshold (described below). Note that the genome-skims compared may use different filtering schemes yet Eqn. 5 holds regardless.

Differing genome lengths. Based on a model where the genomic distance between genomes of different lengths is defined to be confined to the mutations that are falling on homologous sequences, we can drive

$$J = \frac{\eta_1\eta_2 \min(L_1, L_2)(1 - D)^k}{\zeta_1 L_1 + \zeta_2 L_2 - \eta_1\eta_2 \min(L_1, L_2)(1 - D)^k} \ .$$

This computation does not penalize for genome length difference. While a rigorous modeling of evolutionary distance for genomes of different length require sophisticated models of gene gain, duplication, and loss, we take the heuristic approach used by Ondov *et al.* [41] and simply replace $\min(L_1, L_2)$ with $(L_1 + L_2)/2$. This ensures that the estimated distance increases as genome lengths becomes successively more different. This leads us to our final estimate of distance given by:

$$D = 1 - \left(\frac{2(\zeta_1 L_1 + \zeta_2 L_2)J}{\eta_1\eta_2(L_1 + L_2)(1 + J)} \right)^{1/k} \tag{7}$$

2.2 Estimating Coverage

So far we have assumed a perfect knowledge of sequencing depth and error. We will continue to use a *given* constant base error rate ϵ (either known or estimated from Phred scores). However, for genome-skims, the genome length is not known; thus, we need to estimate the coverage in order to apply our distance correction.

The sequencing depth, which is the average number of reads covering a position in the genome, can be estimated from the k-mer coverage profiles. The probability distribution of the number of reads covering a

k-mer is a Poisson r.v. with mean λ, where λ is defined as k-mer coverage. As we look into the histogram data, it is easier to work with counts instead of probabilities. Let M denote the total number of k-mers of length k in the genome, and M_i count the number of k-mers covered by i reads. Thus, for $i \geq 0$, $\mathbb{E}[M_i] = M\frac{\lambda^i}{i!}e^{-\lambda}$. For a given set of reads, we can count the number of times that each k-mer is seen, and assuming zero sequencing error, it equals the number of reads covering that k-mer. Then, we can aggregate the number of k-mers covered by i reads and find M_i for $i \geq 1$. However, since in a genome-skim, large parts of the genome may not be covered, both M and M_0 are unknown. To deal with this issue, we could take the ratio of consecutive counts to get a series of estimates of λ as $\tilde{\lambda}_i = \frac{M_{i+1}}{M_i}(i+1)$ for $i = 1, 2, \ldots$. In practice, sequencing errors change the frequency of k-mers which has to be considered when estimating the coverage. Like before, we assume that the k-mer length k is large enough that any error will introduce a novel k-mer, so the count of all erroneous k-mers is added to the count of single-copy k-mers. Moreover, for k-mers with more than one copy, the number of times that each kmer is seen equals the number of reads covering that k-mer without any error. Formally, let $\hat{M}_i$ denote the count of k-mers seen i times in the presence of error, and $\rho = (1-\epsilon)^k$ denote the probability of error-free k-mer.

$$\mathbb{E}[\hat{M}_i] = \begin{cases} \sum_{j=i} M\frac{\lambda^j}{j!}e^{-\lambda}\binom{j}{i}\rho^i(1-\rho)^{j-i} & i \geq 2 \\ \sum_{j=1} M\frac{\lambda^j}{j!}e^{-\lambda}\left(j\rho(1-\rho)^{j-1} + j(1-\rho)\right) & i = 1 \end{cases} = \begin{cases} M\frac{(\lambda\rho)^i}{i!}e^{-\lambda\rho} & i \geq 2 \\ M\left(\lambda\rho e^{-\lambda\rho} + \lambda(1-\rho)\right) & i = 1 \end{cases}$$

If we know the error rate, then λ can be estimated using the information in $\hat{M}_i$'s. Similar to the case of zero error, a family of estimates is obtained by taking the ratio of consecutive counts

$$\tilde{\lambda}_i = \begin{cases} f^{-1}(\frac{2\hat{M}_2}{\hat{M}_1}(1-\rho)) & i = 1 \\ \frac{1}{\rho}\frac{\hat{M}_{i+1}}{\hat{M}_i}(i+1) & i \geq 2 \end{cases} \tag{8}$$

where $f(\lambda) = \lambda\rho^2 e^{-\lambda\rho} - \frac{2\hat{M}_2}{\hat{M}_1}(\rho e^{-\lambda\rho})$. For the case of $i = 1$, we solve the equation numerically, starting from $\tilde{\lambda}_2$. While any of these $\tilde{\lambda}_i$ can be used in principle, the empirical performance can be affected by the choice; in our tool, we use heuristic rules (described below) that seek to use error-free but large M_i values.

3 Experimental setup

Skmer takes as input two or more genome-skims and a point estimate of sequencing error, ϵ. It uses JellyFish [42] to compute M_i values, which are then used in estimating λ based on Equation 8. We first compute $h = \text{argmax}_{i\geq 2} M_i$; if $c = \frac{l}{l-k}\tilde{\lambda}_h > 4$, $\lambda = \tilde{\lambda}_{h+1}$; otherwise, if $2 \leq c \leq 4$, $\lambda = \tilde{\lambda}_h$; finally, if $c < 2$, $\lambda = \tilde{\lambda}_1$. Then, Mash is used to estimate the Jaccard index, as described below. Finally, we use Equation 7 to compute the hamming distance with η and ζ values computed using Equations 3, 4 if $c < 5$ or else using Equation 6. Also, the genome length L is estimated as the total sequence length divided by the coverage c.

We used a series of experiments to (i) study the accuracy of our new approach compared to existing methods with respect to computing the hamming distance, and (ii) finding the reference match to a query sequence in a reference dataset of genome-skims, or the closest match when the query is not included in the reference.

We compared performance against *Mash/Mash** and *AAF*[30]. For Mash, and Skmer, we used $k = 31$ (selected empirically; Fig. S1) and sketch size 10^7. As Mash handles errors by removing low copy k-mers,

we set the minimum cardinality for k-mers to be included as $\lfloor \frac{c}{5} \rfloor + 1$ with our estimate of c. We also created a version of Mash called Mash* that did not use the approximation $(1-D)^k \simeq e^{-kD}$. AAF [30] is another method that uses k-mers to estimate distances. AFF has an algorithm to correct hamming distances for low coverage, but the correction relies on adjusting the length of tip branches in a distance-based inferred phylogeny. As such, it cannot run on a pair of genomes and requires at least four genomes. Also, AAF leaves coverage estimation to the user with some guidelines, which we fully follow (Appendix C).

Genomic Datasets. We used three sets of publicly available assembled genomes (Tables S3–S5) and used ART [44] to simulate genome-skims, controlling for the sequencing depth (coverage) and introducing sequencing error at a fixed rate of $\epsilon = 0.01$ (Appendix C). Specifically, the data included 21 *Drosophila* genomes (flies) and 22 genomes from the *Anopheles* genus (mosquitoes) obtained from InsectBase[45], and 47 avian species from the Avian Phylogenomic Project [46, 47]. We also used simulations to control mutation distance between pairs of genomes. As a challenging case, we took the highly repetitive assembly of the wasp species *Cotesia vestalis*, and mutated it artificially; we only applied single nucleotide mutations distributed uniformly at random across the genome. We repeated the study on the simpler case of the fly species *D. melanogaster*. Similar to real genomes, we generate genome-skims using ART with $\epsilon = 0.01$ and varying coverage between $\frac{1}{64}$X and 16X. For simulated genomes, we repeated the skimming 10 times and reported the mean and standard error.

Evaluation Metrics. For simulated data, the true distance is controlled and is thus known. For biological datasets, the ground truth is unknown. Instead, we use the distance measured on the full assembly by each method as its ground truth; thus, the ground truth for AAF is computed using AAF. We show both absolute error and the relative error, measured as $|\frac{d-\hat{d}}{d}|$ where d and $\hat{d}$ are the true and the estimated distances.

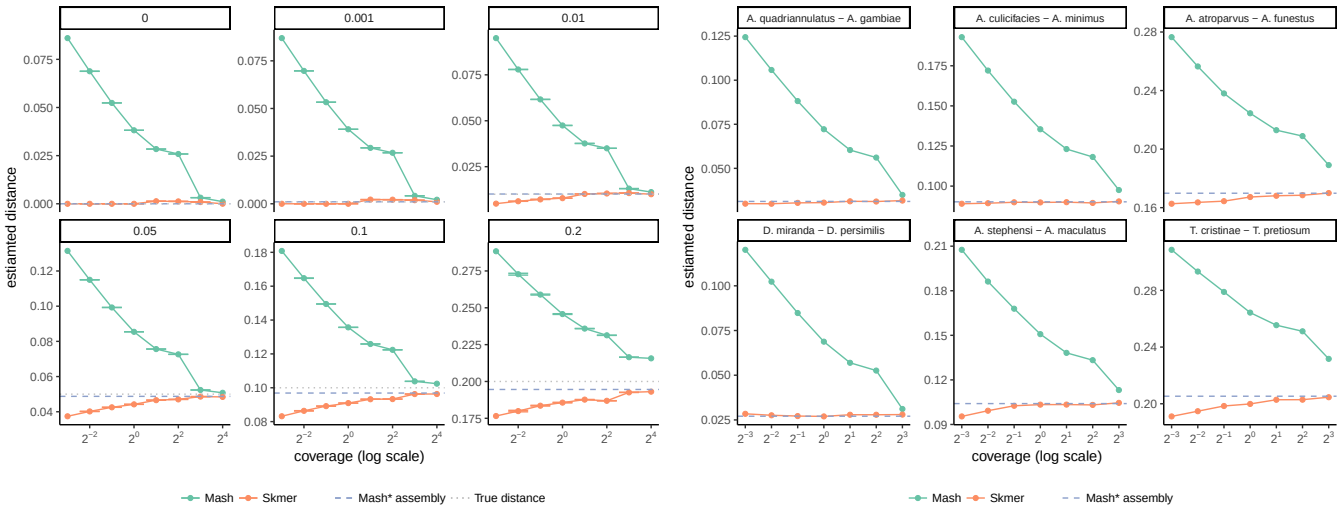
Leave-i-out. We used a leave-i-out strategy to study the accuracy of searching for a query genome in a reference set. For a query genome G_q in a set of n genomes $\{G_1 \dots G_n\}$, we ordered all genomes based on their distances to G_q calculated using the full assemblies, which represents the ground truth; let $G_q^1 \dots G_q^n$ denote the order (note $G_q^1 = G_q$). For $1 < i < n$, we removed the closest $i-1$ genomes to G_q from the reference dataset, leaving us with $G_q^i \dots G_q^n$. We then ordered the remaining genomes by each method; let $x_1 \dots x_{n-i+1}$ be the order obtained by a method and let r be the the rank of the best remaining genome according to the grand truth in the estimated order (i.e., $x_r = G_q^i$). Since $r = 1$ implies perfect performance, and $r > 1$ indicates error, we measured error as the mean of $r-1$ across all query genomes ($1 \le q \le n$).

4 Results

4.1 Hamming distance for pairs of genome-skims

We first study the accuracy of Mash and Skmer in estimating the hamming distance between a pair of genomes. Since AAF cannot be run on pairs of genomes, we do not test it in our first set of analyses.

Simulated Genomes. On simulated genomes, where we control both the distance and coverage, distances are computed with high accuracy by Mash when coverage is high (Fig. 1a), except where the true distance is also high (i.e., 0.2). However, the accuracy of Mash quickly degrades when the coverage is reduced to 4X or less. In contrast, even when the coverage is reduced to $\frac{1}{8}$X, Skmer has high accuracy. For example, with the true distance set to 0.05, Mash estimates the distance as 0.085 with 1X coverage (an overestimation by



(a) *C. vestalis* v.s. a simulated genome

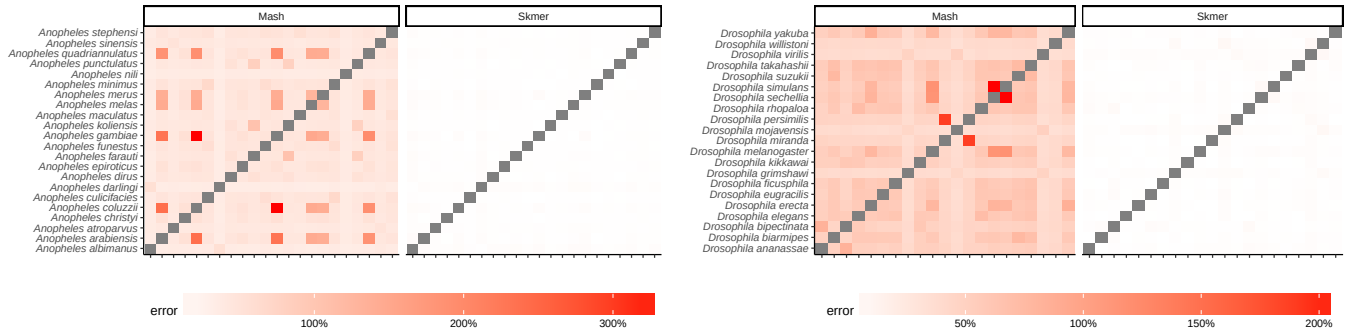
(b) Pairs of real insect genomes

Figure 1: **Comparing the distances estimated by Mash and Skmer.** Hamming distance between the genome-skims simulated using ART with read length $\ell = 100$, constant base error rate $\epsilon = 0.01$, and varying range of coverage (x-axis). a) Six pairs of genomes simulated by applying substitutions to the assembly of *C. vestalis*. The mean and standard error of distances are shown over 10 repeats. b) Six pairs of insects genomes at different hamming distances. The pairs placed in the top (bottom) row consist of species with similar (different) genome lengths. Dashed line (Mash* run on the assembly) is taken as the ground truth.

70%) while Skmer corrects the distance to 0.044 (an underestimation by 12%). Note that applying Mash* to the complete assemblies generally generates very accurate results, as expected, but even given the full assembly, Mash* still has a small but noticeable error when $d = 0.2$. We note that repeating skimming ten times with different samples produces extremely consistent estimates. Repeating the process with the *Drosophila melanogaster* genome as the base genome also produces similar results (Fig. S2). The only condition where Skmer has considerable absolute error is with coverage below 1X and $d = 0.2$ (Fig. 1a). However, we note that for $d = 0.001$, the relative error is not small with low coverage (Fig. S3b) indicating that distinguishing very small distances (perhaps below species-level) requires high coverage. Estimating the right order of magnitude when the true distance is 0.001 seems to require at least 2X coverage while 1X coverage is sufficient to distinguish distances at or above 0.01 (Fig. S3).

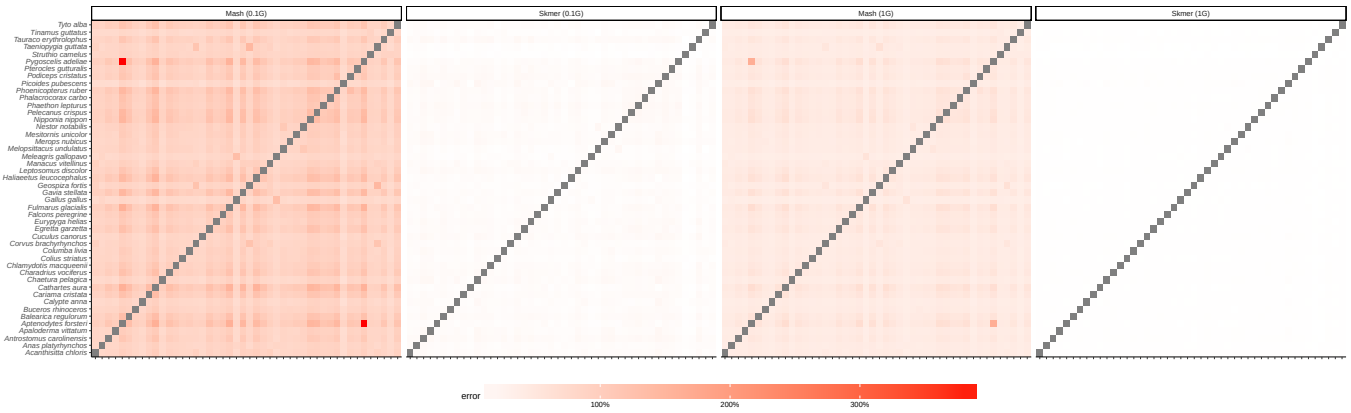
To find the minimum levels of coverage required for accurately estimating the hamming distance using Skmer, we repeat the simulation but range the coverage from $\frac{1}{64}$X to 1X (Fig. S4). Interestingly, even with very low coverage, the absolute error in estimated distances is relatively small, especially when the true distance is also small (for $d \geq 0.1$, Skmer estimates start to degrade below $\frac{1}{8}$X coverage).

Real Genomes. We now test methods on real pairs of insect and avian genomes. Note that unlike the simulated datasets, here, genomes can undergo all types of genetic variations and complex rearrangements, and thus, do not have the same length. Since the true distance cannot be controlled, we carefully selected several pairs of genomes to cover a wide range of mutation distance and genome length. Here, the distance estimated by Mash* on the assemblies is considered the true distance. For all pairs of insect and avian genomes (Fig. 1b and Fig. S5), Mash has high error for coverage below 8X while Skmer successfully corrects the estimated distance and obtains values extremely close to the the results of running Mash* on the full assembly. For example, the distance between *A. stephensi* with length ∼196Mbp and *A. maculatus* with



(a) *Anopheles*

(b) *Drosophila*



(c) Birds

Figure 2: **Comparing the error of Mash and Skmer in distance estimation with fixed amount of sequence from each species.** (a) The dataset of 22 *Anopheles* genomes, skimmed with 0.1Gb sequence. See Figure S7 for 0.5Gb and 1Gb cases. (b) The dataset of 21 *Drosophila* genomes, skimmed with 0.1Gb sequence. See Figure S8 for 0.5Gb and 1Gb cases. (c) The dataset of 47 avian genomes, skimmed with 0.1Gb and 1Gb sequence. See Figure S9 for the 0.5Gb case. The error of Mash for the two eagle species (*H. albicilla* and *H. leucocephalus*) was extremely large, dominating the color spectrum; we excluded *H. albicilla* to help with readability. For instance, with 0.1Gb, Mash estimates the distance between the eagles to be 0.114, which is >40 times larger than the true distance (0.0027); Skmer's estimate is 0.0018 (∼30% error).

length ∼132Mbp is estimated to be 0.104 based on the full assembly and 0.103 (1% underestimation) with only $\frac{1}{2}$X coverage using Skmer, while Mash would estimate the distance to be 0.168 (60% overestimation). Interestingly, on real data, Skmer seems to have even less error than simulated genomes.

Coverage estimates. Our estimates of c are close to the true c used in simulations (Fig. S6a). Notably, Skmer run with the true coverage is *less* accurate than with estimated coverage (Fig. S6b). We speculate that on genomes with repeats, by slightly overestimating coverage, our method gives an estimate of the "effective" coverage, reducing the impact of repeats on the Jaccard index.

4.2 Sets of genome-skims

We now turn to datasets with sets of genome-skims. So far, our experiments have controlled for the coverage by skimming varying amount of sequence data, proportional to the genome length. In our genome-skimming application, coverage will not be fixed. Often, the amount of sequence data obtained for each species will be relatively similar. As a result, genomes of different length end up being sequenced with different coverage

Table 1: **Comparing the average error of Mash and Skmer over three datasets.** Fixed sequencing effort from each species.

Dataset	Sequencing effort	Mash	Skmer
Anopheles	0.1Gb	51.57% (1.72%)	2.46% (0.07%)
	0.5Gb	28.41% (0.74%)	0.82% (0.02%)
	1Gb	20.38% (0.71%)	0.39% (0.02%)
Drosophila	0.1Gb	52.14% (0.94%)	1.77% (0.07%)
	0.5Gb	29.13% (0.45%)	0.64% (0.02%)
	1Gb	14.56% (0.20%)	0.52% (0.02%)
Birds	0.1Gb	99.41% (2.69%)	7.77% (0.07%)
	0.5Gb	60.86% (1.54%)	2.62% (0.02%)
	1Gb	45.50% (1.10%)	1.68% (0.01%)

* The standard error of the mean is provided in parentheses.

Table 2: **Comparing the average error of Mash, Skmer, and AAF over three datasets.** Mixed sequencing effort

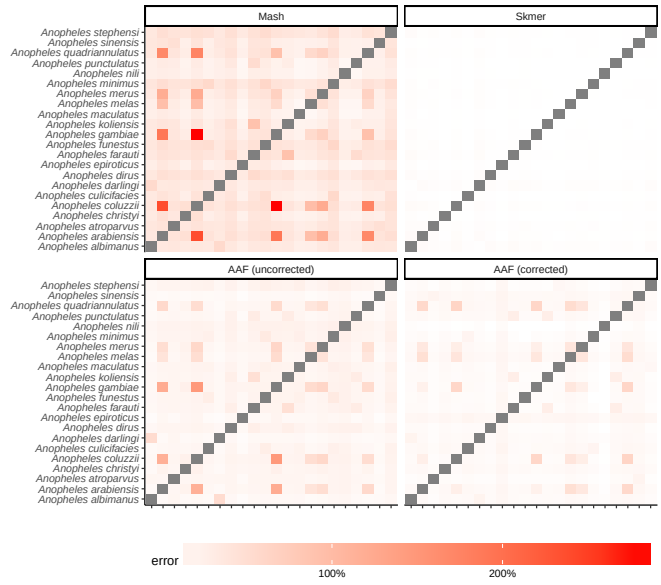
Dataset	Mash	Skmer	AAF (uncorrected)	AAF (corrected)
Anopheles	38.49% (1.43%)	1.36% (0.06%)	18.01% (0.74%)	9.78% (0.45%)
Drosophila	34.29% (0.95%)	0.78% (0.03%)	18.03% (0.72%)	9.33% (0.34%)
Birds	73.41% (1.59%)	3.32% (0.05%)	43.07% (1.93%)	8.71% (0.73%)

* The standard error of the mean is provided in parentheses.

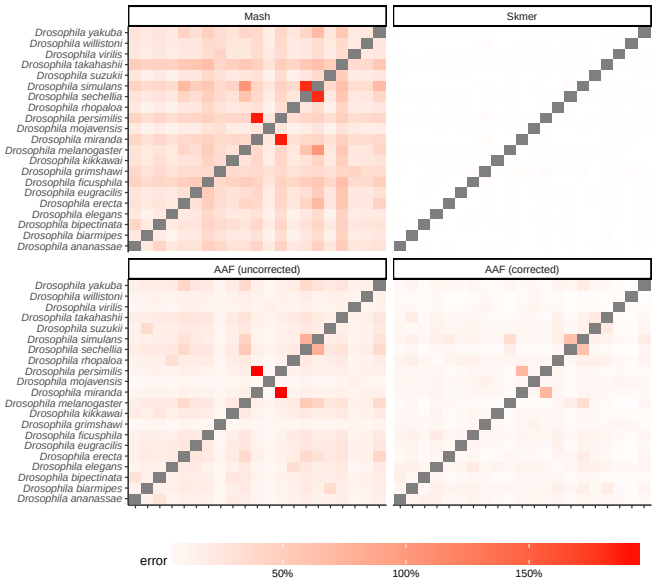
depth proportional to the inverse of their length. Moreover, the sequencing effort per species may also vary across sequencing protocols, experiments and research labs, and so a database of reference genome-skims may consist of samples with heterogeneous sequencing coverages. We now study the accuracy of different methods in the presence of mixed coverage.

Fixed sequencing effort. We start with experiments where all species are skimmed with the same sequencing effort (0.1Gb, 0.5Gb, or 1Gb) and measure the error in the estimated mutation distance between all pairs of species in the *Anopheles*, *Drosophila*, and avian datasets (Figs. 2, S7– S9). The error in the distance estimated by Mash relative to the ground truth can be quite large (higher than 200% in the worst case) while Skmer consistently makes accurate estimates close to the true distance even at the lowest amount of coverage (Table 1). We should note that the typical genome length of species varies among these three datasets, and so equal sequencing effort means unequal range of sequencing coverage. For instance, the birds genomes are on average ∼5 times larger than *Anopheles* genomes; thus, birds need to be skimmed with larger amount of sequence to have an accuracy comparable with *Anopheles* species. As expected, increasing coverage reduces the error for all methods including Mash (Figs. S7– S9)

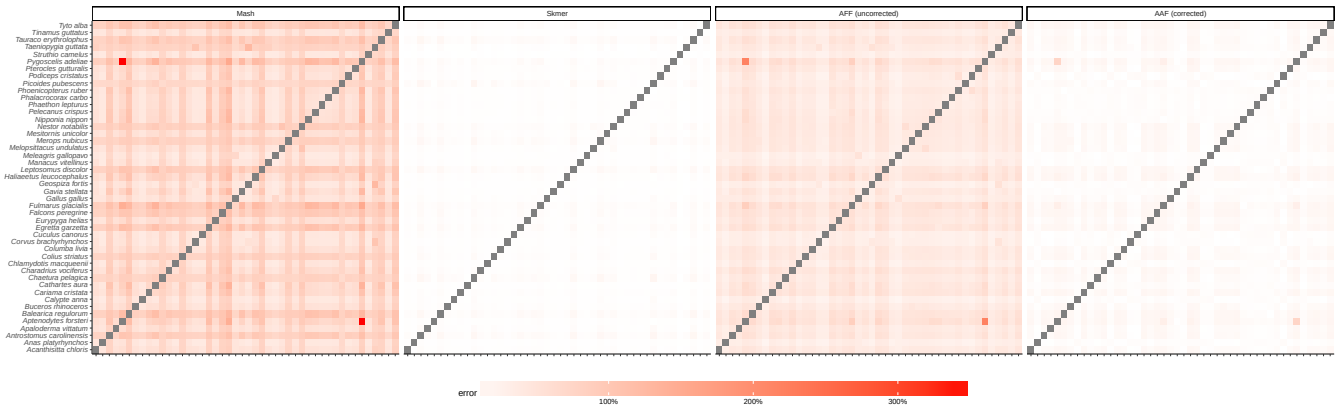
Heterogeneous sequencing effort. We now further mix coverages as follows to capture the scenario where genome-skims come from various labs or experimental protocols. For each species, we choose its total sequencing effort from three possible values 0.1Gb, 0.5Gb, and 1Gb, uniformly at random, and estimate all pairs of distances within each dataset as before (Fig. 3). Here, in addition to Mash, we also compare our results with AAF. Similar to the case of fixed sequencing effort, Skmer mitigates large relative error in the



(a) *Anopheles*



(b) *Drosophila*



(c) Birds

Figure 3: **Comparing the error of Mash, Skmer, and AAF with mixed coverage.** Species have random amount of sequence chosen uniformly among 0.1Gb, 0.5Gb, and 1Gb. (a) *Anopheles* dataset. (b) *Drosophila* dataset. (c) The avian dataset. Similar to (Fig. 2c), we have excluded one of the eagles (*H. albicilla*). The error of Mash, AAF, and Skmer in estimating the distance between the two eagles are 2340%, 1107%, and 6.5%, respectively (both of the eagles are skimmed at 0.5Gb here.) True distance used in calculating the error of AAF and Skmer is computed by applying each method to the genome assemblies.

distances estimated by Mash and produces accurate results. The correction applied by AAF also reduces the impact of low coverage to some extent; still Skmer has considerably less error (Table 2). For example, in the *Drosophila* dataset, the worst-case error of AAF is above 70%, whereas it never exceeds 4% for Skmer.

Sequencing Error. We tested the impact of (i) providing an incorrect estimate of ϵ to Skmer and (ii) using uneven distributions of error that change across the length of the read to emulate the Illumina HiSeq2000 platform. Skmer seems generally robust to mis-specifications of the sequencing error model, especially when the error is underestimated (Fig. 4 and Table S6). However, overestimating the error (e.g., setting it 2% where the true error rate is 1%) leads to a noticeable increase of the distance errors. Using uneven patterns of error across a read has minimal negative impacts on the accuracy of Skmer.

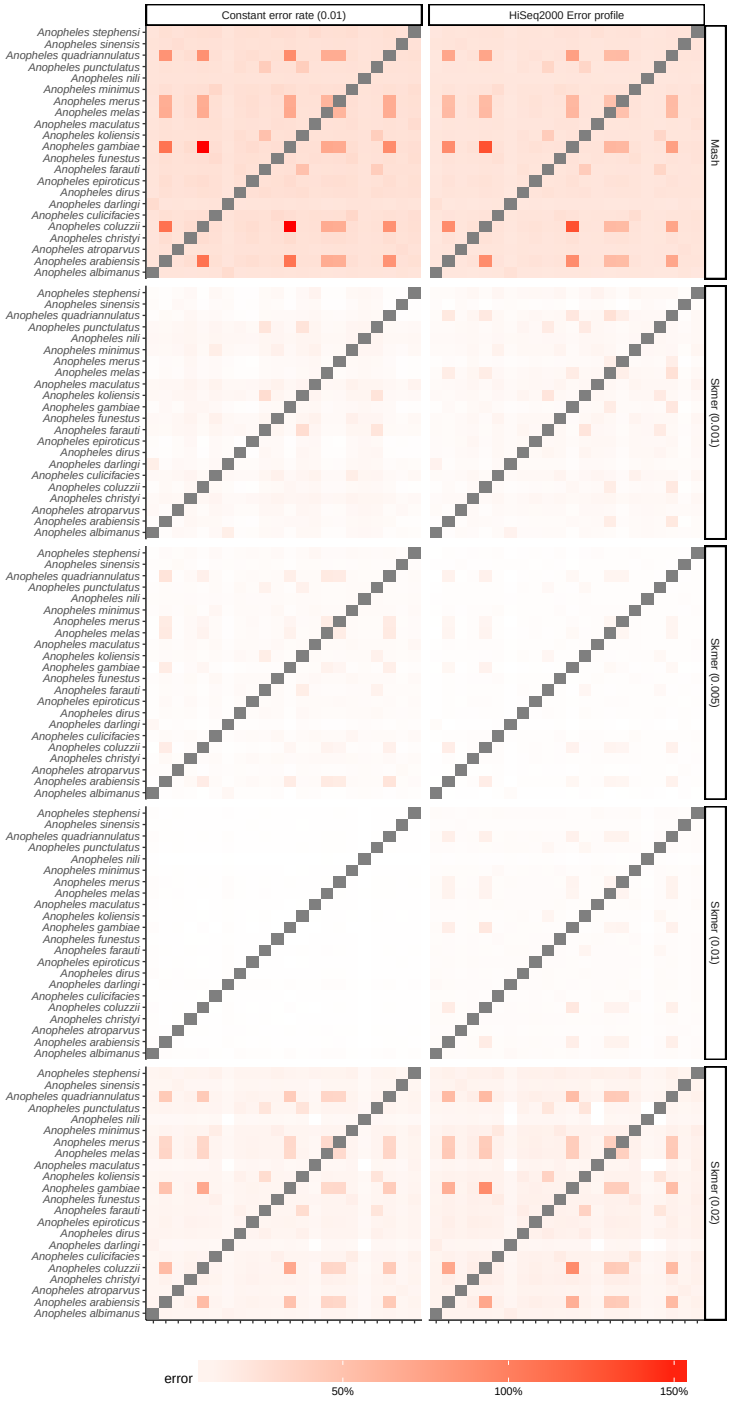
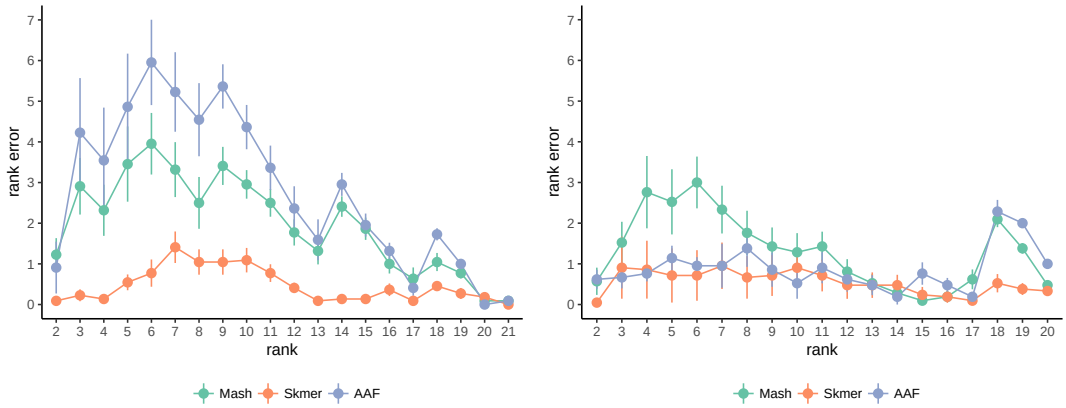


Figure 4: **Distance correction sensitivity to the estimate of sequencing error rate.** Comparing the error of Mash and Skmer on the dataset of 22 *Anopheles* genomes, with 0.5Gb sequence from each species. Genomes skimmed using ART with constant base error rate $\epsilon = 0.01$ (left), and Illumina HiSeq2000 error profile (right). Skmer was run with the estimated ϵ set to 0.001, 0.005, 0.01, and 0.02.

Running time. In terms of running time, Skmer and Mash are comparable while AAF is much slower. For example, the total running time (using 24 CPU cores) to compute distances based on genome-skims for all $\binom{47}{2}$ pairs of birds using Mash, Skmer, and AAF was roughly 8, 33, and 460 minutes, respectively.



(a) *Anopheles* (b) *Drosophila*

Figure 5: **The mean rank error of the best remaining match in leave-i-out experiments.** Comparing Mash, Skmer, and AAF on (a) the *Anopheles* dataset, and (b) the *Drosophila* dataset.

4.3 Leave-out search against a reference database of genome-skims

We now study the effectiveness of using hamming distance to search a database of genome-skims to find the closest match to a query genome-skim. Given a query genome-skim and a reference dataset of genomes, we can order the reference genomes based on their hamming distance to the query. The results can be provided to the user as a ranking. When the query genome is available in the reference dataset, finding the match is relatively easy. To study the effectiveness of the search as the distance of the closest available match increases, we use a leave-out experiment, as described earlier in Section 3. Figure 5 shows the mean rank error of the best remaining match in a leave-out experiment when removing $i-1$ genomes for $1 < i < n$. Recall that rank error zero corresponds to a perfect match to the best available genome.

On the *Anopheles* dataset, Skmer consistently outperforms Mash and AAF in terms of finding the best remaining match. In fact, for finding the second, third, or fourth best match, Skmer has close to zero error. In contrast, Mash and AFF are on average off by one genome even for finding the second best match. On the *Drosophila* datasets, finding matches seems relatively easier for all three methods. Still, in finding the second best match, Skmer again has close to zero error while AAF and Mash are each off by a genome close to half of the times. After the second best match, AAF and Skmer have comparable accuracy while Mash is considerably worse. These results demonstrate that correcting the distance not only impacts our understanding of the absolute distance, but also, impacts estimates of the relative distance of genome-skims.

5 Discussion

We showed that hamming distances as small as 0.01 can be estimated accurately from genome-skims with 1X or lower coverage. What does a distance of 0.01 mean? The answer will depend on the organisms of interest. For example, two eagles species of the same genus (*H. albicilla* and *H. leucocephalus*) have $D \approx 0.003$ but two *Anopheles* species of the same species complex (*A. gambiae* and *A. coluzzii*) have $D \approx 0.018$. Broadly speaking, for eukaryotes, detecting distances in the 10^{-2} order is often enough to distinguish between species (Fig. S10). On the other hand, distances in the 10^{-3} order often differentiate between populations or very similar species. Detection at these lower levels seems to require 2X coverage using Skmer (Fig. S3b) but future work should study the exact level of sequencing required for accurate ordering of species at distances

in the order of 10^{-3} or less. Moreover, the question of the minimum coverage required may avail itself to information-theoretical bounds and near-optimal solutions, similar to those established for the assembly problem [48, 49].

All of our tests in this study were based on simulating genome-skims from assemblies by sub-sampling reads and adding sequencing error. While this provided us with reliable ground truth of distances, real applications of genome-skims may face further complications. For example, the actual coverage of real genome-skims may not be uniform and randomly distributed. At a minimum, actual genome skims will have an overrepresentation of mitochondrial or plastid sequence. Moreover, the read length may be different between the query and the reference genome-skims. More importantly, other sources of DNA originating from for example, parasites, diet, fungi, commensals, bacteria, and human contamination may all be present in the sample and may cause an over-estimation of the distance. This may or may not impact the ranking of a genome skim with regards to the reference species, but it certainly *can* impact the value of the estimated distance. We recommend that before using Skmer, database searches should be used to find and eliminate bacterial or fungal contamination (perhaps using metagenomic tools such as Kraken [50]). Our future efforts will further study ways to eliminate impacts of external DNA. A related direction of future work is to explore whether Skmer can be extended to environmental DNA analyses, i.e., queries consisting of genome-skims of multi-taxa samples. While Skmer is presented here in a general setting, its best use is for eukaryotic organisms, where the notion of species is better established and species can be separated with reasonable effort. We tested Skmer on birds and insects, but we predict it will work equally well for plants, a prediction that should be tested in future work.

The connection between hamming distance and phylogenetic distance depends on mutation processes considered. If only substitutions are allowed and assuming the Jukes-Cantor model [51], the phylogenetic distance is $-\frac{3}{4}\ln(1-\frac{4}{3}d)$; note this transformation is monotonic and does not change rankings of matches to a query search. Assuming a more complex model such as GTR [52], hamming distance is not enough to estimate the phylogenetic distance. However, we have devised a simple procedure to estimate GTR distances using the log-det approach [53] by repeated applications of Skmer to perturbed reads (Appendix B). The GTR distances can rank matches to a query differently from the hamming distance; the accuracy of the two distances should be compared in future work. Insertions, deletions, duplications, losses, and repeats can all reduce the Jaccard index and thus increase the hamming distance. However, with these mutations, the correct definition of the evolutionary distance is not straightforward; nor is its relationship to hamming distance or Jaccard index clear. Here, we focused on estimating the hamming distance with high accuracy despite low coverage, leaving these broader questions to future work.

References

[1] P. D. N. Hebert, A. Cywinska, S. L. Ball, and J. R. deWaard, "Biological identifications through DNA barcodes," *Proceedings of the Royal Society B: Biological Sciences*, vol. 270, no. 1512, pp. 313–321, 2003.

[2] V. Savolainen, R. S. Cowan, A. P. Vogler, G. K. Roderick, and R. Lane, "Towards writing the encyclopaedia of life: an introduction to DNA barcoding," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 360, no. 1462, pp. 1805–1811, 2005.

[3] P. Taberlet, E. Coissac, F. Pompanon, C. Brochmann, and E. Willerslev, "Towards next-generation biodiversity assessment using DNA metabarcoding," *Molecular Ecology*, vol. 21, pp. 2045–2050, apr 2012.

[4] K. A. Seifert, R. A. Samson, J. R. deWaard, J. Houbraken, C. A. Levesque, J.-M. Moncalvo, G. Louis-Seize, and P. D. N. Hebert, "Prospects for fungus identification using CO1 DNA barcodes, with Penicillium as a test case," *Proceedings of the National Academy of Sciences*, vol. 104, no. 10, pp. 3901–3906, 2007.

[5] M. Vences, M. Thomas, A. van der Meijden, Y. Chiari, and D. R. Vieites, "Comparative performance of the 16S rRNA gene in DNA barcoding of amphibians," *Frontiers in zoology*, vol. 2, p. 5, 2005.

[6] A. Ardura, A. R. Linde, J. C. Moreira, and E. Garcia-Vazquez, "DNA barcoding for conservation and management of Amazonian commercial fish," *Biological Conservation*, vol. 143, no. 6, pp. 1438–1443, 2010.

[7] P. M. Hollingsworth, L. L. Forrest, J. L. Spouge, M. Hajibabaei, S. Ratnasingham, M. van der Bank, M. W. Chase, R. S. Cowan, D. L. Erickson, A. J. Fazekas, S. W. Graham, K. E. James, K.-J. Kim, W. J. Kress, H. Schneider, J. van AlphenStahl, S. C. Barrett, C. van den Berg, D. Bogarin, K. S. Burgess, K. M. Cameron, M. Carine, J. Chacon, A. Clark, J. J. Clarkson, F. Conrad, D. S. Devey, C. S. Ford, T. A. Hedderson, M. L. Hollingsworth, B. C. Husband, L. J. Kelly, P. R. Kesanakurti, J. S. Kim, Y.-D. Kim, R. Lahaye, H.-L. Lee, D. G. Long, S. Madrinan, O. Maurin, I. Meusnier, S. G. Newmaster, C.-W. Park, D. M. Percy, G. Petersen, J. E. Richardson, G. A. Salazar, V. Savolainen, O. Seberg, M. J. Wilkinson, D.-K. Yi, and D. P. Little, "A DNA barcode for land plants," *Proceedings of the National Academy of Sciences*, vol. 106, pp. 12794–12797, 8 2009.

[8] C. L. Schoch, K. A. Seifert, S. Huhndorf, V. Robert, J. L. Spouge, C. A. Levesque, W. Chen, E. Bolchacova, K. Voigt, P. W. Crous, A. N. Miller, M. J. Wingfield, M. C. Aime, K.-D. An, F.-Y. Bai, R. W. Barreto, D. Begerow, M.-J. Bergeron, M. Blackwell, T. Boekhout, M. Bogale, N. Boonyuen, A. R. Burgaz, B. Buyck, L. Cai, Q. Cai, G. Cardinali, P. Chaverri, B. J. Coppins, A. Crespo, P. Cubas, C. Cummings, U. Damm, Z. W. de Beer, G. S. de Hoog, R. Del-Prado, B. Dentinger, J. Dieguez-Uribeondo, P. K. Divakar, B. Douglas, M. Duenas, T. A. Duong, U. Eberhardt, J. E. Edwards, M. S. Elshahed, K. Fliegerova, M. Furtado, M. A. Garcia, Z.-W. Ge, G. W. Griffith, K. Griffiths, J. Z. Groenewald, M. Groenewald, M. Grube, M. Gryzenhout, L.-D. Guo, F. Hagen, S. Hambleton, R. C. Hamelin, K. Hansen, P. Harrold, G. Heller, C. Herrera, K. Hirayama, Y. Hirooka, H.-M. Ho, K. Hoffmann, V. Hofstetter, F. Hognabba, P. M. Hollingsworth, S.-B. Hong, K. Hosaka, J. Houbraken, K. Hughes, S. Huhtinen, K. D. Hyde, T. James, E. M. Johnson, J. E. Johnson, P. R. Johnston, E. B. G. Jones, L. J. Kelly, P. M. Kirk, D. G. Knapp, U. Koljalg, G. M. Kovacs, C. P. Kurtzman, S. Landvik, S. D. Leavitt, A. S. Liggenstoffer, K. Liimatainen, L. Lombard, J. J. Luangsa-ard, H. T. Lumbsch, H. Maganti, S. S. N. Maharachchikumbura, M. P. Martin, T. W. May, A. R. McTaggart, A. S. Methven, W. Meyer, J.-M. Moncalvo, S. Mongkolsamrit, L. G. Nagy, R. H. Nilsson, T. Niskanen, I. Nyilasi, G. Okada, I. Okane, I. Olariaga, J. Otte, T. Papp, D. Park, T. Petkovits, R. Pino-Bodas, W. Quaedvlieg, H. A. Raja, D. Redecker, T. L. Rintoul, C. Ruibal, J. M. Sarmiento-Ramirez, I. Schmitt, A. Schussler, C. Shearer, K. Sotome, F. O. P. Stefani, S. Stenroos, B. Stielow, H. Stockinger, S. Suetrong, S.-O. Suh, G.-H.

Sung, M. Suzuki, K. Tanaka, L. Tedersoo, M. T. Telleria, E. Tretter, W. A. Untereiner, H. Urbina, C. Vagvolgyi, A. Vialle, T. D. Vu, G. Walther, Q.-M. Wang, Y. Wang, B. S. Weir, M. Weiss, M. M. White, J. Xu, R. Yahr, Z. L. Yang, A. Yurkov, J.-C. Zamora, N. Zhang, W.-Y. Zhuang, and D. Schindel, "Nuclear ribosomal internal transcribed spacer (ITS) region as a universal DNA barcode marker for Fungi," *Proceedings of the National Academy of Sciences*, vol. 109, pp. 6241–6246, apr 2012.

[9] D.-s. Zhang, Y.-d. Zhou, C.-s. Wang, and G. Rouse, "A new species of Ophryotrocha (Annelida, Eunicida, Dorvilleidae) from hydrothermal vents on the Southwest Indian Ridge," *ZooKeys*, vol. 687, pp. 1–9, 8 2017.

[10] M. C. Hedin and W. P. Maddison, "A Combined Molecular Approach to Phylogeny of the Jumping Spider Subfamily Dendryphantinae (Araneae: Salticidae)," *Molecular Phylogenetics and Evolution*, vol. 18, pp. 386–403, 3 2001.

[11] K. H. Taylor, G. W. Rouse, and C. G. Messing, "Systematics of Himerometra (Echinodermata: Crinoidea: Himerometridae) based on morphology and molecular data," *Zoological Journal of the Linnean Society*, vol. 181, pp. 342–356, 10 2017.

[12] S. Ratnasingham and P. D. N. Hebert, "BOLD : The Barcode of Life Data System (www.barcodinglife.org)," *Molecular Ecology Notes*, vol. 7, no. April 2016, pp. 355–364, 2007.

[13] D. Steinke, M. Vences, W. Salzburger, and A. Meyer, "TaxI: a software tool for DNA barcoding using distance methods," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 360, no. 1462, pp. 1975–1980, 2005.

[14] S. Mirarab, N. Nguyen, and T. Warnow, "SEPP: SATé-Enabled Phylogenetic Placement.," *Pacific Symposium On Biocomputing*, pp. 247–58, 2012.

[15] S. A. Berger, D.K., A. Stamatakis, and D. Krompass, "Performance, Accuracy, and Web Server for Evolutionary Placement of Short Sequence Reads under Maximum Likelihood," *Systematic Biology*, vol. 60, pp. 291–302, 5 2011.

[16] F. A. Matsen, R. B. Kodner, and E. V. Armbrust, "pplacer: linear time maximum-likelihood and Bayesian phylogenetic placement of sequences onto a fixed reference tree.," *BMC bioinformatics*, vol. 11, p. 538, 1 2010.

[17] M. J. Hickerson, C. P. Meyer, C. Moritz, and M. Hedin, "DNA Barcoding Will Often Fail to Discover New Animal Species over Broad Parameter Space," *Systematic Biology*, vol. 55, pp. 729–739, 10 2006.

[18] D. L. J. Quicke, M. Alex Smith, D. H. Janzen, W. Hallwachs, J. Fernandez-Triana, N. M. Laurenne, A. Zald??var-River??n, M. R. Shaw, G. R. Broad, S. Klopfstein, S. R. Shaw, J. Hrcek, P. D. N. Hebert, S. E. Miller, J. J. Rodriguez, J. B. Whitfield, M. J. Sharkey, B. J. Sharanowski, R. Jussila, I. D. Gauld, D. Chesters, and A. P. Vogler, "Utility of the DNA barcoding gene fragment for parasitic wasp phylogeny (Hymenoptera: Ichneumonoidea): Data release and new measure of taxonomic congruence," *Molecular Ecology Resources*, vol. 12, pp. 676–685, 7 2012.

[19] E. Coissac, P. M. Hollingsworth, S. Lavergne, and P. Taberlet, "From barcodes to genomes: Extending the concept of DNA barcoding," 2016.

[20] S. C. K. Straub, M. Parks, K. Weitemier, M. Fishbein, R. C. Cronn, and A. Liston, "Navigating the tip of the genomic iceberg: Next-generation sequencing for plant systematics," *American Journal of Botany*, vol. 99, pp. 349–364, feb 2012.

[21] "France Génomique - Mutualisation des compétences et des équipements français pour l'analyse génomique et la bio-informatique." `https://www.france-genomique.org/`.

[22] "Norwegian Barcode of Life (NorBOL)." `http://www.norbol.org/en/`.

[23] "DNAmark." `http://dnamark.ku.dk/english/`.

[24] J. Tonti-Filippini, P. G. Nevill, K. Dixon, and I. Small, "What can we do with 1000 plastid genomes?," *Plant Journal*, vol. 90, no. 4, pp. 808–818, 2017.

[25] B. E. Blaisdell, "A measure of the similarity of sets of sequences not requiring sequence alignment.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 83, pp. 5155–9, 7 1986.

[26] S. Vinga and J. Almeida, "Alignment-free sequence comparison–a review," *Bioinformatics*, vol. 19, pp. 513–523, 3 2003.

[27] G. Reinert, D. Chew, F. Sun, and M. S. Waterman, "Alignment-free sequence comparison (I): statistics and power.," *Journal of computational biology : a journal of computational molecular cell biology*, vol. 16, pp. 1615–34, 12 2009.

[28] J. L. Thorne and H. Kishino, "Freeing phylogenies from artifacts of alignment.," *Molecular biology and evolution*, vol. 9, pp. 1148–62, 11 1992.

[29] M. Höhl and M. A. Ragan, "Is multiple-sequence alignment required for accurate inference of phylogeny?," *Systematic Biology*, vol. 56, no. 2, pp. 206–221, 2007.

[30] H. Fan, A. R. Ives, Y. Surget-Groba, and C. H. Cannon, "An assembly and alignment-free method of phylogeny reconstruction from next-generation sequencing data," *BMC Genomics*, vol. 16, p. 522, 7 2015.

[31] C. Daskalakis and S. Roch, "Alignment-free phylogenetic reconstruction: Sample complexity via a branching process analysis," *Annals of Applied Probability*, vol. 23, no. 2, pp. 693–721, 2013.

[32] Q. Dai, Y. Yang, and T. Wang, "Markov model plus k-word distributions: a synergy that produces novel statistical measures for sequence comparison," *Bioinformatics*, vol. 24, pp. 2296–2302, 10 2008.

[33] K. Yang and L. Zhang, "Performance comparison between k-tuple distance and four model-based distances in phylogenetic tree reconstruction," *Nucleic Acids Research*, vol. 36, pp. e33–e33, 1 2008.

[34] J. Qi, H. Luo, and B. Hao, "CVTree: a phylogenetic tree reconstruction tool based on whole genomes," *Nucleic Acids Research*, vol. 32, pp. W45–W47, 7 2004.

[35] I. Ulitsky, D. Burstein, T. Tuller, and B. Chor, "The Average Common Substring Approach to Phylogenomic Reconstruction," *Journal of Computational Biology*, vol. 13, pp. 336–350, 3 2006.

[36] H. Yi and L. Jin, "Co-phylog: an assembly-free phylogenomic approach for closely related organisms," *Nucleic Acids Research*, vol. 41, pp. e75–e75, 4 2013.

[37] T. Roychowdhury, A. Vishnoi, and A. Bhattacharya, "Next-Generation Anchor Based Phylogeny (NexABP): Constructing phylogeny from Next-generation sequencing data," *Scientific Reports*, vol. 3, p. 2634, 12 2013.

[38] K. Song, J. Ren, Z. Zhai, X. Liu, M. Deng, and F. Sun, "Alignment-Free Sequence Comparison Based on Next-Generation Sequencing Reads," *Journal of Computational Biology*, vol. 20, pp. 64–79, 2 2013.

[39] M. Domazet-Lošo and B. Haubold, "Alignment-free detection of local similarity among viral and bacterial genomes," *Bioinformatics*, vol. 27, pp. 1466–1472, 6 2011.

[40] B. Haubold, "Alignment-free phylogenetics and population genetics," *Briefings in Bioinformatics*, vol. 15, pp. 407–418, 5 2014.

[41] B. D. Ondov, T. J. Treangen, P. Melsted, A. B. Mallonee, N. H. Bergman, S. Koren, and A. M. Phillippy, "Mash: fast genome and metagenome distance estimation using MinHash," *Genome Biology*, vol. 17, p. 132, 12 2016.

[42] G. Marçais and C. Kingsford, "A fast, lock-free approach for efficient parallel counting of occurrences of k-mers," *Bioinformatics*, vol. 27, pp. 764–770, 3 2011.

[43] Y. Fofanov, Y. Luo, C. Katili, J. Wang, Y. Belosludtsev, T. Powdrill, C. Belapurkar, V. Fofanov, T.-B. Li, S. Chumakov, and B. M. Pettitt, "How independent are the appearances of n-mers in different genomes?," *Bioinformatics*, vol. 20, pp. 2421–2428, 10 2004.

[44] W. Huang, L. Li, J. R. Myers, and G. T. Marth, "ART: a next-generation sequencing read simulator," *Bioinformatics*, vol. 28, pp. 593–594, 2 2012.

[45] C. Yin, G. Shen, D. Guo, S. Wang, X. Ma, H. Xiao, J. Liu, Z. Zhang, Y. Liu, Y. Zhang, K. Yu, S. Huang, and F. Li, "InsectBase: a resource for insect genomes and transcriptomes," *Nucleic Acids Research*, vol. 44, pp. D801–D807, 1 2016.

[46] E. D. Jarvis, S. Mirarab, A. J. Aberer, B. Li, P. Houde, C. Li, S. Y. W. Ho, B. C. Faircloth, B. Nabholz, J. T. Howard, A. Suh, C. C. Weber, R. R. da Fonseca, J. Li, F. Zhang, H. Li, L. Zhou, N. Narula, L. Liu, G. Ganapathy, B. Boussau, M. S. Bayzid, V. Zavidovych, S. Subramanian, T. Gabaldón, S. Capella-Gutiérrez, J. Huerta-Cepas, B. Rekepalli, K. Munch, M. H. Schierup, B. Lindow, W. C. Warren, D. Ray, R. E. Green, M. W. Bruford, X. Zhan, A. Dixon, S. Li, N. Li, Y. Huang, E. P. Derryberry, M. F. Bertelsen, F. H. Sheldon, R. T. Brumfield, C. V. Mello, P. V. Lovell, M. Wirthlin, M. P. C. Schneider, F. Prosdocimi, J. A. Samaniego, A. M. V. Velazquez, A. Alfaro-Núñez, P. F. Campos, B. Petersen, T. Sicheritz-Ponten, A. Pas, T. Bailey, P. Scofield, M. Bunce, D. M. Lambert, Q. Zhou, P. Perelman, A. C. Driskell, B. Shapiro, Z. Xiong, Y. Zeng, S. Liu, Z. Li, B. Liu, K. Wu, J. Xiao, X. Yinqi, Q. Zheng, Y. Zhang, H. Yang, J. Wang, L. Smeds, F. E. Rheindt, M. J. Braun, J. Fjeldså, L. Orlando, F. K. Barker, K. A. Jønsson, W. Johnson, K.-P. Koepfli, S. O'Brien, D. Haussler, O. A. Ryder, C. Rahbek, E. Willerslev, G. R. Graves, T. C. Glenn, J. E. McCormack, D. W. Burt, H. Ellegren, P. Alström, S. V. Edwards, A. Stamatakis, D. P. Mindell, J. Cracraft, E. L. Braun, T. Warnow, W. Jun, M. T. P.

Gilbert, and G. Zhang, "Whole-genome analyses resolve early branches in the tree of life of modern birds," *Science*, vol. 346, pp. 1320–1331, 12 2014.

[47] E. D. Jarvis, S. Mirarab, A. J. Aberer, B. Li, P. Houde, C. Li, S. Y. W. Ho, B. C. Faircloth, B. Nabholz, and J. T. Howard, "Phylogenomic analyses data of the avian phylogenomics project," *GigaScience*, vol. 4, no. 1, p. 4, 2015.

[48] G. Bresler, M. Bresler, and D. Tse, "Optimal assembly for high throughput shotgun sequencing.," *BMC bioinformatics*, vol. 14 Suppl 5, no. Suppl 5, p. S18, 2013.

[49] I. Shomorony, S. H. Kim, T. A. Courtade, and D. N. C. Tse, "Information-optimal genome assembly via sparse read-overlap graphs," *Bioinformatics*, vol. 32, no. 17, pp. i494–i502, 2016.

[50] D. E. Wood and S. L. Salzberg, "Kraken: ultrafast metagenomic sequence classification using exact alignments," *Genome Biology*, vol. 15, no. 3, p. R46, 2014.

[51] T. H. Jukes and C. R. Cantor, "Evolution of protein molecules," in *In Mammalian protein metabolism, Vol. III (1969), pp. 21-132*, vol. III, pp. 21–132, 1969.

[52] S. Tavaré, "Some Probabilistic and Statistical Problems in the Analysis of DNA Sequences," *Lectures on Mathematics in the Life Sciences*, vol. 17, pp. 57–86, 1986.

[53] P. Erdos, M. Steel, L. Szekely, and T. Warnow, "A few logs suffice to build (almost) all trees: Part II," *Theoretical Computer Science*, vol. 221, no. 1-2, pp. 77–118, 1999.

Supplementary Material

A Theoretical results

Consider two genomes of identical length L and separated by hamming distance D where the hamming distance is defined as the fraction of variant sites between the perfect alignment of the two genomes. We would like to estimate D from two genome-skims.

Mutations. We model the two genomes as the outcome of a random process that copies a genome and introduces mutations at each position i.i.d with a fixed probability d. Indexing from left to right, we can define $n = L-k+1$ k-mers (note that $n \approx L$ for any reasonable choice of k and genome length). Let X_i be a binary random variable (r.v.) that indicates whether k-mer i is identical between the two genomes. Clearly, in our model, $X_i \sim \text{Bern}(p)$ where $p = (1-d)^k$. Then, $W = \sum_1^n X_i$ gives the number of shared k-mers. If J is defined as the Jaccard index over the set of all k-mers from both genomes, it's easy to see that $J = \frac{W}{2n-W}$ and thus, $\frac{W}{n} = \frac{2J}{1+J}$. We further make a simplifying assumption. We assume all X_i r.v.s are independent, an assumption that is true for most pairs of k-mers but ignores the fact that each k-mer overlaps with k-1 other k-mers. With this assumption, the maximum likelihood estimate of p is simply

$$\hat{p} = \frac{W}{n} = \frac{2J}{1+J} \ .$$

By the functional invariance of maximum likelihood, the ML estimate of d is given by:

$$\hat{d} = 1 - \left(\frac{2J}{1+J}\right)^{\frac{1}{k}} \ .$$

k-mer sampling. We now assume that each genome is covered uniformly at random. Thus, k-mers are also sub-sampled and we assume each k-mer is sampled at least once with probability η_1 in the first genome and η_2 in the second genome; we derive the relationship between these probabilities and genome coverage below. We estimate η values separately (also described below) and here consider them as given. For each $1 \leq i \leq n$ and $j \in \{1, 2\}$, let $Y_{j,i} \sim \text{Bern}(\eta_j)$ be the indicator of whether the k-mer i is sampled at least once in the genome j. Under this scenario, the number of k-mers shared between the two genomes is given by the r.v. $W = \sum_1^n X_i Y_{1,i} Y_{2,i}$. Defining $Z = X_i Y_{1,i} Y_{2,i}$, we get $W = \sum_1^n Z_i$ and $Z_i \sim \text{Bern}(r)$ where $r = p\eta_1\eta_2$ by the independence of the mutation process and each of the two k-mer sampling processes. Assuming independence between Z_i r.v.s (again ignoring the overlap between consecutive k-mers) we get the ML estimate $\hat{r} = \frac{W}{n}$, and thus (for a given η_1 and η_2) we have

$$\hat{r} = \hat{p}\eta_1\eta_2 = \frac{W}{n} \tag{S1}$$

Let $U = \sum_1^n S_i$ where $S_i = Y_{1,i} + Y_{2,i} - Y_{1,i}Y_{2,i}X_i$. It is easy to see that U gives the total number of sampled k-mers in both genomes. However, S_i is not a Bernoulli and thus, U is not Binomial. Nevertheless, the same assumptions that we used to treat X_i and Z_i r.v.s as independent also give us independence between S_i values; therefore, by the central limit theorem, $\frac{U}{n}$ can be approximated by a Gaussian with mean $q = \mathbb{E}[S_i]$. Moreover, $\mathbb{E}[S_i] = \mathbb{E}[Y_{1,i}] + \mathbb{E}[Y_{2,i}] - \mathbb{E}[Y_{1,i}Y_{2,i}X_i] = \eta_1 + \eta_2 - \eta_1\eta_2 p$ (note that X_i, $Y_{1,i}$ and $Y_{2,i}$ are independent). By this Gaussian approximation, the ML estimate of q given η_1, η_2 is given by:

$$\hat{q} = \eta_1 + \eta_2 - \eta_1\eta_2\hat{p} = \frac{U}{n} \ . \tag{S2}$$

Note that $J = \frac{W}{U}$. Equations S1 and S2 give two different ML estimators of the same parameter p given two different types of data (W and U). While the two estimators are not the same, because n is extremely large, both estimators have a very low variance. Exploiting the low variance, we treat the two estimates of p as equal and divide both sides of Equation S1 by Equation S2 to get:

$$\frac{\hat{r}}{\hat{q}} = \frac{W}{U} = J = \frac{\hat{p}\eta_1\eta_2}{\eta_1 + \eta_2 - \eta_1\eta_2\hat{p}} .$$

Solving for $\hat{p}$ and replacing $\hat{d} = 1 - \hat{p}^{\frac{1}{k}}$ gives

$$\hat{d} = 1 - \left(\frac{(\eta_1 + \eta_2)J}{\eta_1\eta_2(1 + J)}\right)^{\frac{1}{k}} .$$

Note that we have assumed a known coverage and thus we are not co-estimating η_j's and d. In practice, we need to first estimate η_1 and η_2, and we do it as we will describe.

Connection of η to read coverage. A k-mer stretching from position y to $y+k$ on the genome is covered by the reads that start in the interval $[y+k-\ell, y]$. Assuming that there is no sequencing error, and a uniform spread of of the N reads across the genome of length L. We show that the probability η that a k-mer is sampled by at least one read is given by

$$\eta = 1 - e^{-c(1-\frac{k}{\ell})}$$

Let X be a r.v. denoting the number of reads that cover a specific k-mer. Assuming a uniform spread of N reads across the genome of length L, the probability of x reads covering a k-mer (starting in an interval of length $\ell - k$) is given by

$$Prob(X = x) = \binom{N}{x}\left(\frac{l-k}{L}\right)^x\left(1 - \frac{l-k}{L}\right)^{N-x}$$

As N is large and $\frac{N(l-k)}{L}$ is constant, it can be closely approximated by

$$Prob(X = x) = \frac{\lambda^x}{x!}e^{-\lambda}$$

where $\lambda = \frac{N(l-k)}{L}$ is the *k-mer* coverage, and is related to the coverage c by

$$\lambda = \frac{l-k}{l}c$$

As the number of reads covering a k-mer follows Poisson distribution, the fraction of k-mers covered by 1 or more reads is

$$\eta = 1 - e^{-\lambda} \tag{S3}$$

Sequencing error. We model the sequencing error as an i.i.d process that corrupts each position of each read with a fixed probability ϵ. To extend our previous results to cover this scenario, we need to see how the intersection r.v. (W) and the union r.v. (U) get affected.

We start with the intersection (W). We change the meaning of η to denote the probability that a k-mer

is covered by at least one error-free read. The probability of a k-mer within a read being error-free is clearly

$$\rho = (1-\epsilon)^k \simeq e^{-k\epsilon} \tag{S4}$$

By conditioning on the number of reads covering a k-mer, the probability of not covering a k-mer with an error-free read is given by

$$\begin{aligned} Prob(\text{no error-free read}) &= \sum_{i=0}^{\infty} Prob(\text{all reads have error}|i \text{ reads})\, Prob(i \text{ reads}) \\ &= \sum_{i=0}^{\infty} (1-\rho)^i\, Prob(i \text{ reads}) \\ &= \sum_{i=0}^{\infty} (1-\rho)^i \frac{\lambda^i}{i!} e^{-\lambda} \\ &= e^{-\lambda\rho} \end{aligned} \tag{S5}$$

Hence, the probability that a k-mer is covered by at least one error-free read is given by

$$\eta = 1 - e^{-\lambda\rho} \tag{S6}$$

Note that Eqn. S6 reduces to Eqn. S3 when there is no sequencing error, i.e., $\rho = 1$. Similar to the case of no error, given η_1 and η_2, the r.v. $\frac{W}{n}$ (where W is the number of shared k-mers) can be used with Equation S1 to estimate r.

We now turn to the union (r.v. U). For large enough k, and for genomes that are random and repeat-free, with high probability ($> 1 - \frac{2L}{4^k}$) an error produces a new k-mer that is not observed in either of the input genomes. Ignoring the exceedingly unlikely event that two errors produce the same k-mer or that they produce a k-mer present in one of the two genomes, we can assume that the sequencing error generates as many new k-mers as the number of reads being affected by errors.

In the regime that includes errors, $U = \sum_1^n (T_{1,i} + T_{2,i}) - W$ where the r.v.s $T_{1,i}$ and $T_{2,i}$ give the total number of k-mers generated from the position i from the first and second genomes, respectively. W.l.o.g, consider $T_{1,i}$. By conditioning on the number of reads covering a k-mer we have

$$\mathbb{E}[T_{1,i}] = \mathbb{E}[\mathbb{E}[T_{1,i}|x \text{ reads}]] = \sum_{x=0}^{\infty} \mathbb{E}[T_{1,i}|x \text{ reads}]\, Prob(x \text{ reads}) \tag{S7}$$

Given that x reads are covering a k-mer, $T_{1,i}$ equals the number of erroneous k-mers E, plus 1 if there is any error-free k-mer. As $E \sim Binom(x, 1-\rho)$

$$\begin{aligned} \mathbb{E}[T_{1,i}|x \text{ reads}] &= \sum_{j=0}^{x} (j + \mathbf{1}_{j\neq x}) \binom{x}{j} (1-\rho)^j \rho^{x-j} \\ &= x(1-\rho) + (1-(1-\rho)^x) \end{aligned} \tag{S8}$$

and substituting into (S7)

$$\begin{aligned}
\mathbb{E}[T_{1,i}] &= \sum_{x=0}^{\infty}((1-(1-\rho)^x)+x(1-\rho))\,Prob(x \text{ reads}) \\
&= \sum_{x=0}^{\infty}((1-(1-\rho)^x)+x(1-\rho))\frac{\lambda_1^x}{x!}e^{-\lambda_1} \\
&= 1-e^{-\lambda_1\rho}+\lambda_1(1-\rho) \\
&= \eta_1+\lambda_1(1-\rho) \\
&= \eta_1+\lambda_1(1-(1-\epsilon)^k)
\end{aligned} \tag{S9}$$

Letting $\zeta_1 = \mathbb{E}[T_{1,i}]$ and using the same central limit argument we used before, $\frac{U}{n}$ becomes approximately a Gaussian with expectation $\zeta_1+\zeta_2-\eta_1\eta_2 p$. Similar to Equation S2, given ζ_1, ζ_2, η_1, and η_2, the Gaussian approximation gives us:

$$\zeta_1+\zeta_2-\eta_1\eta_2\hat{p} = \frac{U}{n}\,. \tag{S10}$$

Again, assuming that estimates of p in Equation S1 (with the new definition of η) and Equation S10 are the same (due to low variance), we divide the two equations and solve for d to get the estimator:

$$D = 1-\left(\frac{(\zeta_1+\zeta_2)J}{\eta_1\eta_2(1+J)}\right)^{1/k}\,.$$

Excluding low-copy k-mers from the Jaccard index calculation. If we discard k-mers observed less than m times, then a k-mer will survive if it is covered by m or more error-free reads. Hence, η becomes the probability of m or more error-free reads covering a k-mer

$$\begin{aligned}
\eta &= 1-\sum_{t=0}^{m-1} Prob(t \text{ error-free read}) \\
&= 1-\sum_{t=0}^{m-1}\sum_{i=t}^{\infty} Prob(t \text{ error-free read}|i \text{ reads})\,Prob(i \text{ reads}) \\
&= 1-\sum_{t=0}^{m-1}\sum_{i=t}^{\infty}\binom{i}{t}p^t(1-p)^{i-t}\frac{\lambda^i}{i!}e^{-\lambda} \\
&= 1-\sum_{t=0}^{m-1}\frac{(\lambda p)^t}{t!}e^{-\lambda p}
\end{aligned} \tag{S11}$$

In general, we have shown that the probability distribution of the number of error-free k-mers is a Poisson with parameter λp.

B Computing GTR distances

To compute the GTR matrix using the log-det approach, we need a 4×4 matrix F where each element is the fraction of sites where one genome has one letter while the other genome has the other letter. Given this matrix, $d = -\log(\det(F))$ [53].

As elsewhere, we assume a no-indel scenario so that each k-mer mismatch can be attributed to a single nucleotide substitution. For $i, j \in \{\text{A,C,G,T}\}$, let $x_{ij} = x_{ji}$ denote the number of mutations of the form $i \leftrightarrow j$. Our goal is to estimate x_{ij} for all i, j. However, the paradigm of computing distance by hashing/sketching k-mers treats all mutations alike. Formally, the estimated distance d equals

$$d = x_{\text{AC}} + x_{\text{AG}} + x_{\text{AT}} + x_{\text{CG}} + x_{\text{CT}} + x_{\text{GT}}$$

We do the following:

1. Replace G and T with C, and compute distance $d_{\text{A}} = x_{\text{AC}} + x_{\text{AG}} + x_{\text{AT}}$.
2. Replace G and T with A, and compute distance $d_{\text{C}} = x_{\text{AC}} + x_{\text{CG}} + x_{\text{CT}}$.
3. Replace G with T, and compute distance $d_{\text{AC}} = x_{\text{AC}} + x_{\text{AG}} + x_{\text{AT}} + x_{\text{CG}} + x_{\text{CT}}$.

Combining, we get

$$x_{\text{AC}} = d_{\text{A}} + d_{\text{C}} - d_{\text{AC}}$$

A similar procedure can be used to compute all x_{ij} and normalization gives us F.

Note that this procedure reduces the space of possible k-mers of length k to 2^k possibilities instead of 4^k. Therefore, it will likely be required that k is increased for high accuracy when this approach is used.

C Supplementary method details and commands

Here we provide the exact procedures and commands that we used to run external softwares throughout our experiments.

Simulating genome-skims using ART. To simulate short reads with length $\ell = 100$ and constant error rate $\epsilon = 0.01$ (Phred score = 20) at coverage c, we used

```
art_illumina -i FASTA_FILE -l 100 -qL 20 -qU 20 -f c -o FASTQ_FILE
```

To simulate reads with the error profiles of Illumina HiSeq2000, we ran

```
art_illumina -i FASTA_FILE -l 100 -f c -o FASTQ_FILE
```

Computing k-mer frequencies using JellyFish. To count all k-mers of length $k = 31$ in a genome-skim, we used

```
jellyfish count -m 31 -s 100M -C -o COUNT_FILE FASTQ_FILE
```

and to get the histogram of k-mer counts

```
jellyfish histo COUNT_FILE
```

Computing Jaccard index and estimating distance using Mash. We first *sketch* input genome-skims or assemblies with k-mer length $k = 31$ and sketch size $s = 10^7$. For genome-skims (in FASTQ format) when no k-mer filtering is applied, we run

```
mash sketch -r -k 31 -s 10000000 -o SKETCH_FILE FASTQ_FILE
```

To sketch genome-skims while filtering k-mers with less than C copies, we use

```
mash sketch -m C -k 31 -s 10000000 -o SKETCH_FILE FASTQ_FILE
```

For genome assemblies (in FASTA format), we used

```
mash sketch -k 31 -s 10000000 -o SKETCH_FILE FASTA_FILE
```

Then, the Jaccard index and Mash distance between sketches is computed by running

```
mash dist SKETCH_FILE_1 SKETCH_FILE_2
```

Estimating distances using AAF. To count the k-mers ($k = 31$) in a dataset of genome-skims using 24 cores and 120GB memory, we first ran

```
python PATH_to_FILE/aaf_phylokmer.py -k 31 -t 24 -o KMER_COUNT_FILE -d INPUT_DIR -G 120
```

Next, to get the (uncorrected) distances and phylogeny, we used

```
python PATH_to_FILE/aaf_distance.py -i KMER_COUNT_FILE -t 24 -G 120 \
       -o OUTPUT_FILE_PREFIX -f KMER_DIVERSITY_FILE
```

where `KMER_DIVERSITY_FILE` is an output of previous command. Finally, to correct tip branches of phylogeny tree for low coverage and sequencing error, we used

```
python PATH_to_FILE/aaf_tip.py -i TREE_FILE -k 31 --tip TIP_INFO_FILE \
       -f KMER_DIVERSITY_FILE
```

where we had to provide `TIP_INFO_FILE` containing estimates of coverage and sequencing error. To estimate coverage, we followed the procedure suggested in AAF user manual. We first used JellyFish to find the k-mer counts M_i's as described before. They suggest when there is a clear peak in the k-mer frequency distribution, estimate k-mer coverage λ to be the maximum bin. As they do not suggest a specific rule for that, we first find $j = \text{argmax}_{i>1} M_i$, excluding the count of the first bin M_1, which is always large because of erroneous k-mers due to sequencing error. If $j > 2$, it means that we can see a peak in k-mers distribution at j, so we use $\lambda = j$. Otherwise, if $j = 2$, we follow their suggested formula $\lambda = \frac{\sum iM_i}{\sum M_i}$ for the case of low coverage or high sequencing error that there is no clear peak in the k-mer frequency distribution. We should also mention that no k-mer filtering used for AAF, as the coverage was heterogeneous over genome-skims. In fact, in AAF the filtering is applied to all genome-skims if used, and so they suggest to not apply filtering when there is any taxon with low coverage ($c < 5$) within the dataset.

D Supplementary figures and tables

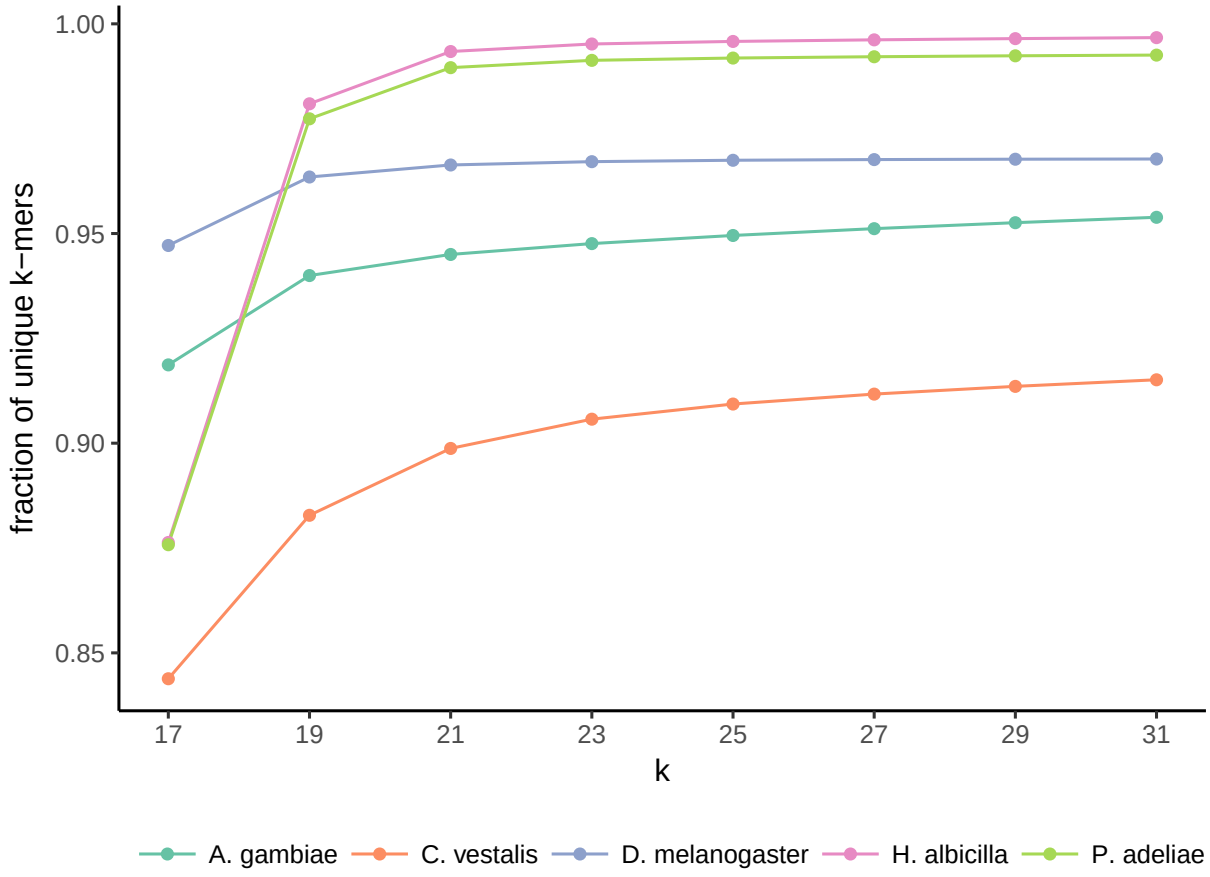


Figure S1: **The fraction of unique k-mers in selected species of insects and birds**

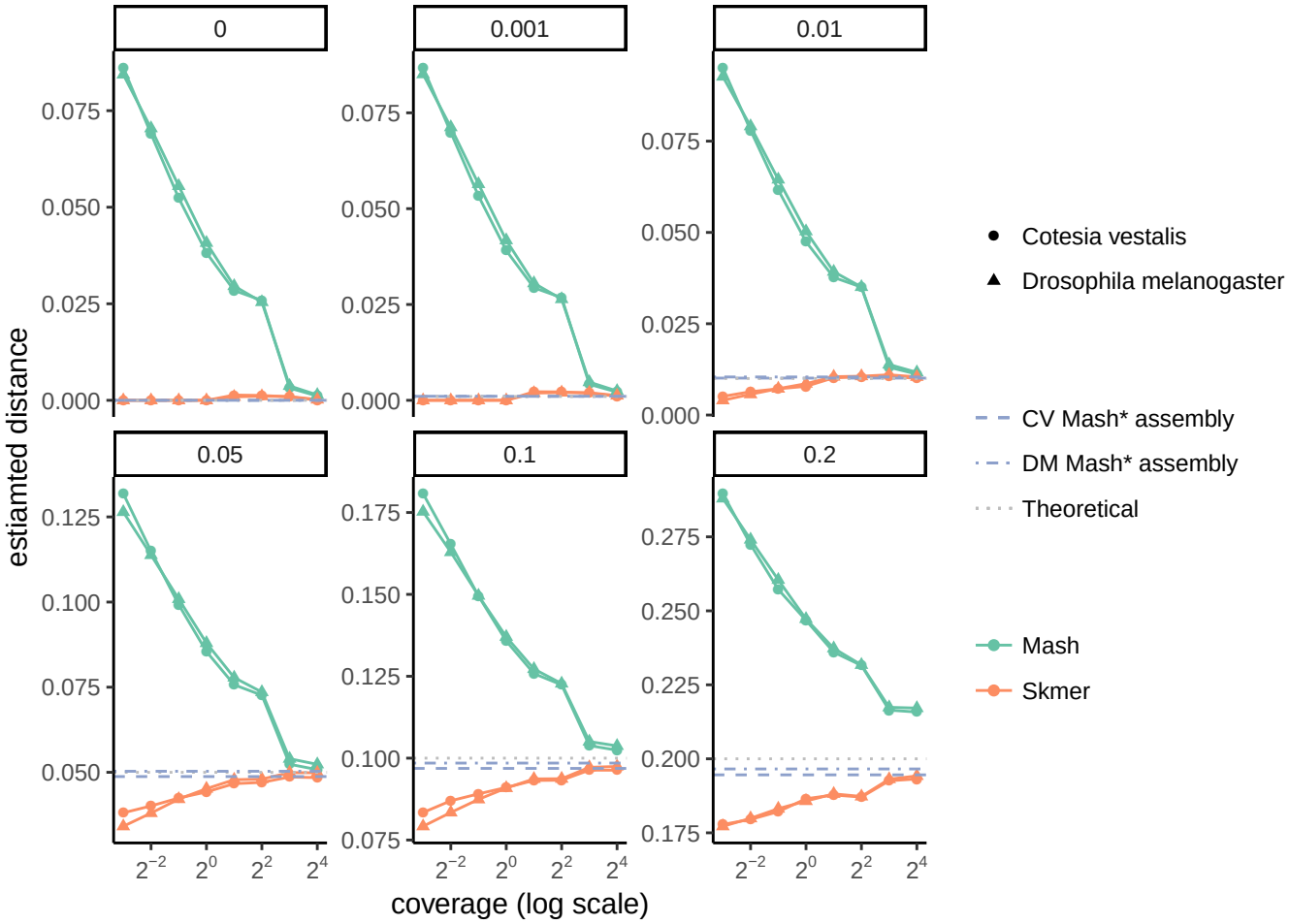
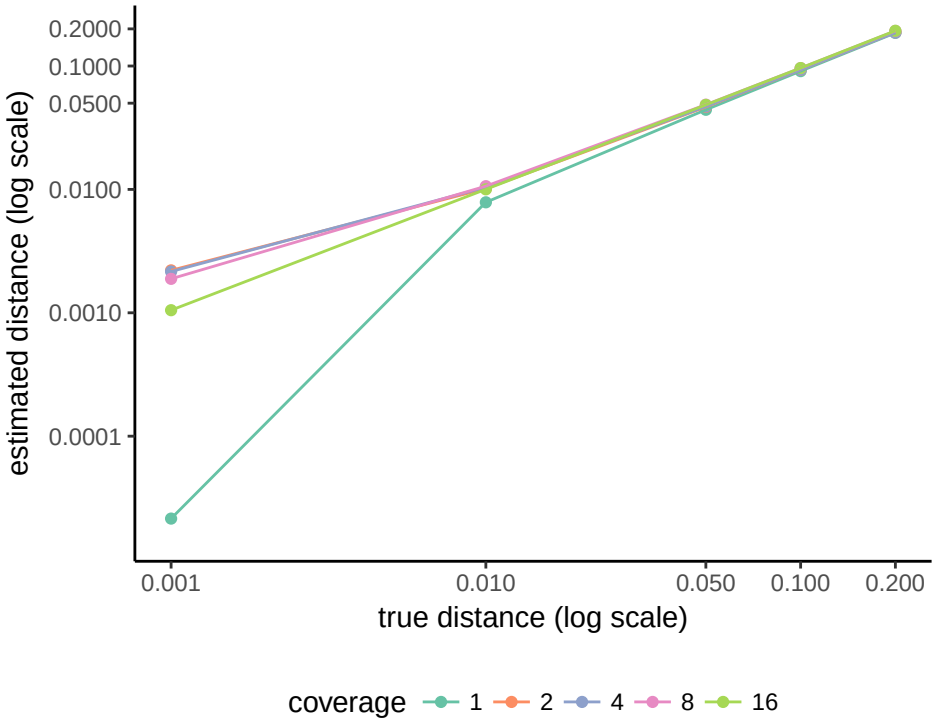
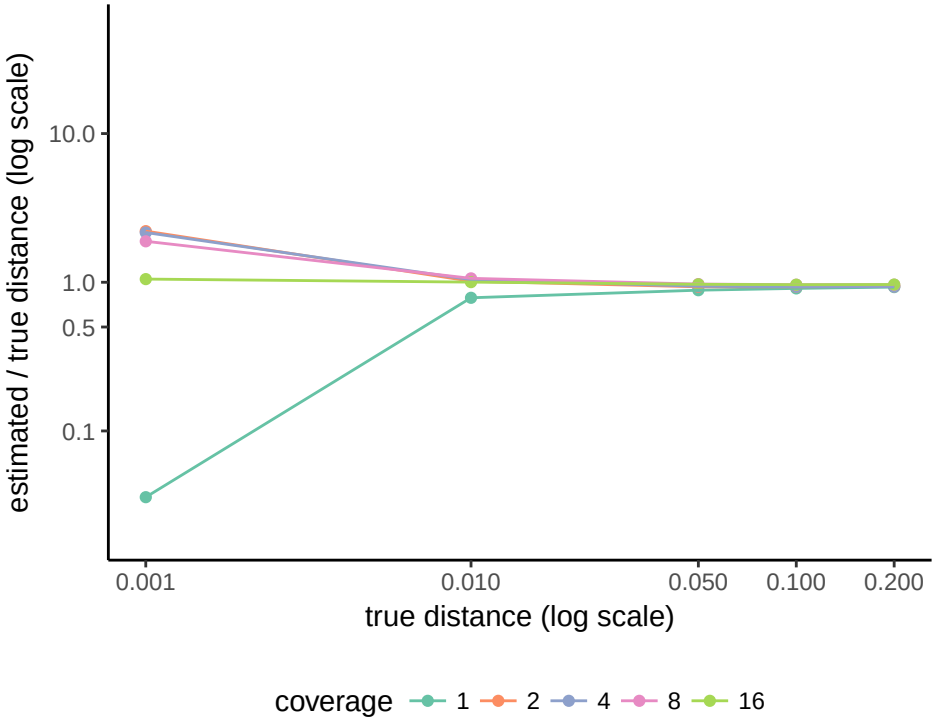


Figure S2: **Comparing distances estimated for genome-skims of two different species.** Genomes simulated at different distances from the genomes of C. vestalis and D. melanogaster and skimmed at a range of coverage from $\frac{1}{8}$X to 16X.



(a)



(b)

Figure S3: **The resolution of Skmer at different hamming distances.** Skims of *C. vestalis* v.s. genomes simulated to be at different distances from *C. vestalis*, with varying coverage. (a) Estimated distance versus the true distance. (b) The ratio of estimated distance to the true distance.

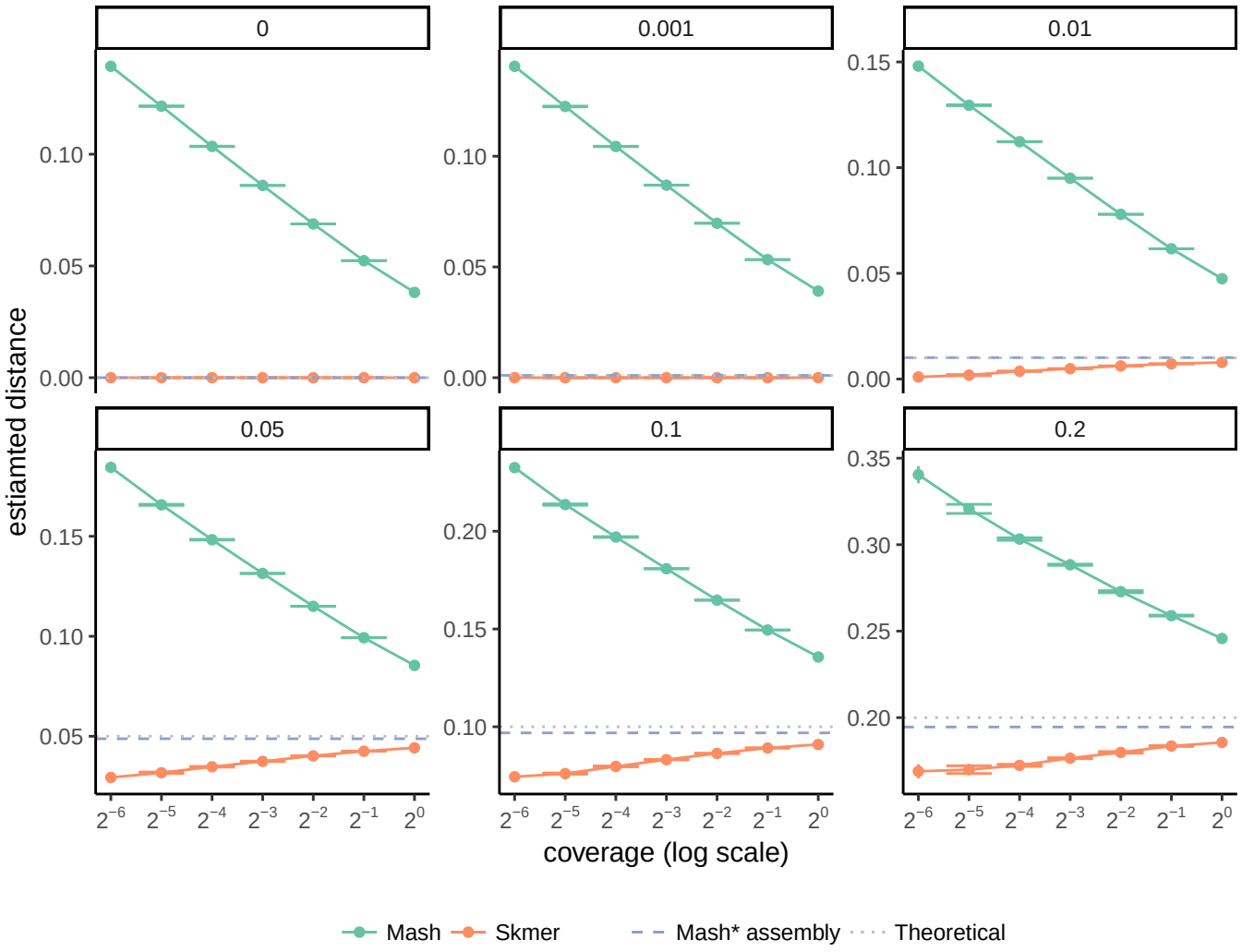


Figure S4: **Comparing distances estimated by Mash and Skmer for simulated data at very low coverages.** Skims of *C. vestalis* v.s. genomes simulated to be at different distances from *C. vestalis*, with varying coverage. The mean and standard error of distances are shown over 10 repeats of the experiment. The coverage ranges from $\frac{1}{64}$X to 1X.

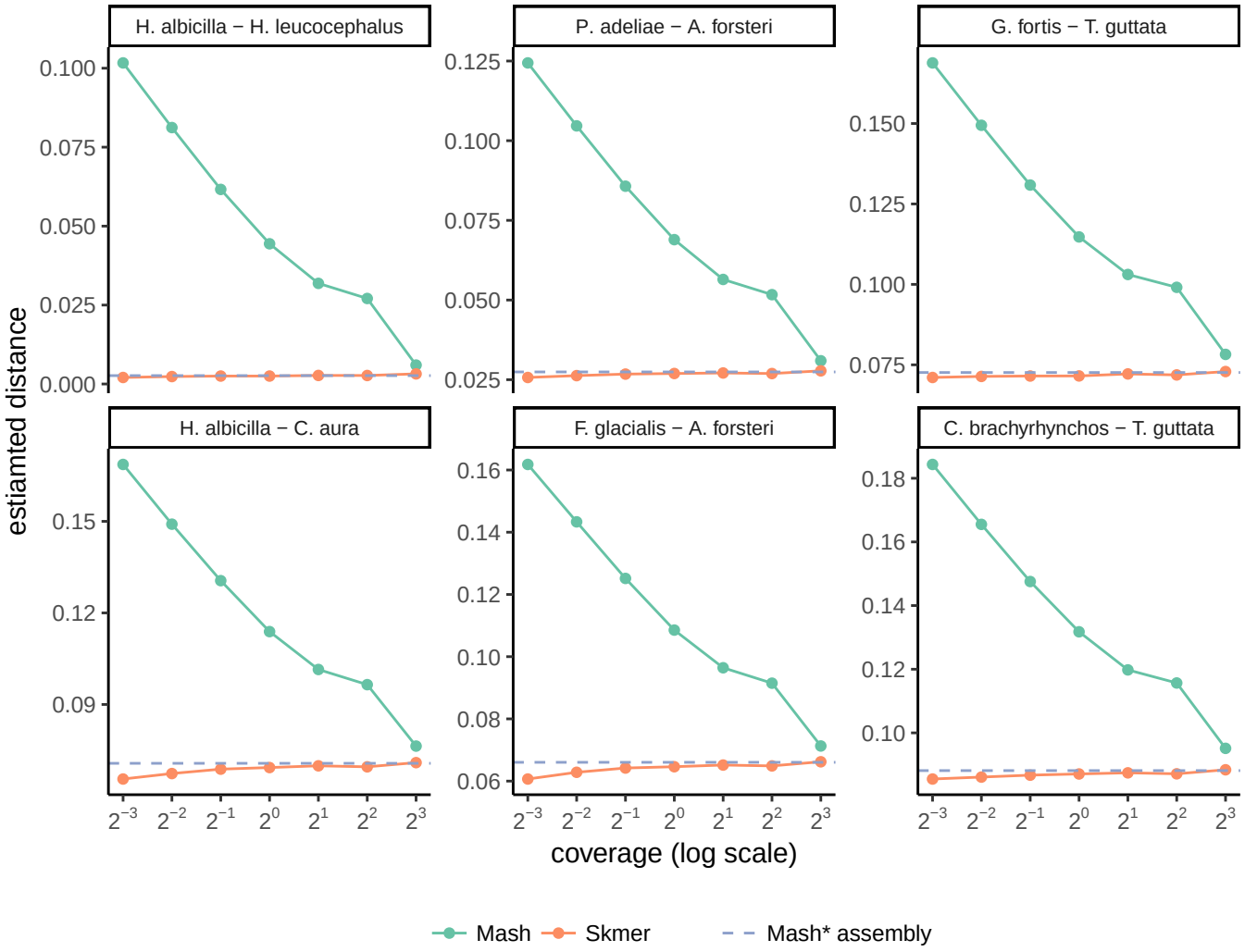
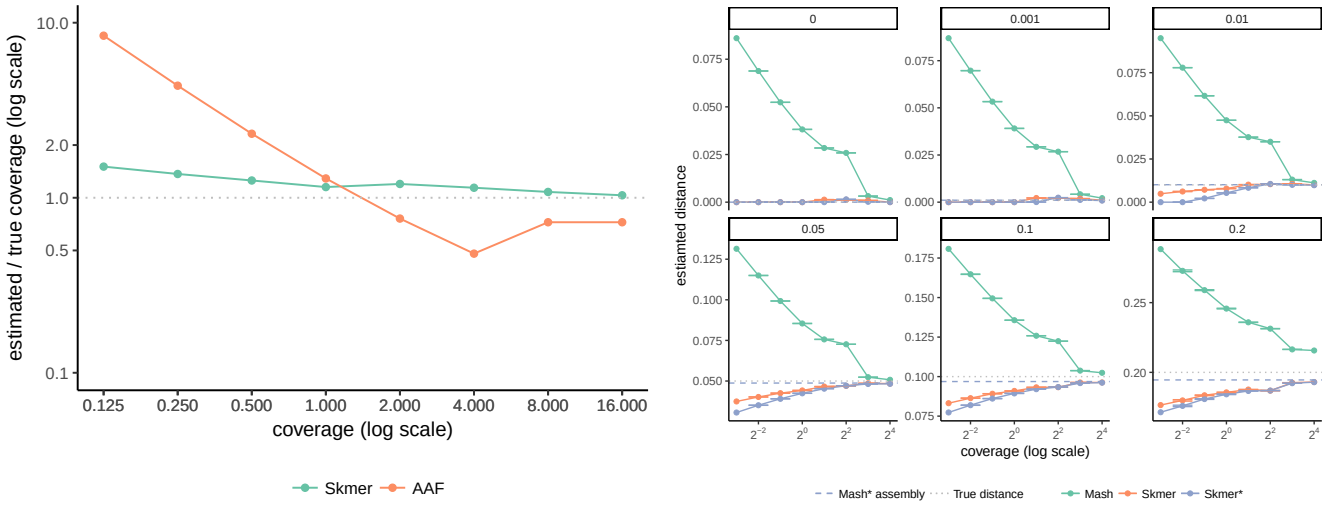


Figure S5: **Comparing distances estimated by Mash and Skmer for real data.** Six pairs of birds genomes at different hamming distances. The coverage ranges from $\frac{1}{8}$X to 8X. Genome lengths are similar for all pairs.



(a) (b)

Figure S6: **The performance of Skmer coverage estimation.** (a) The ratio of estimated coverage to the true coverage (used in simulating genome-skims) for Skmer and AAF. (b) Comparing distances estimated by Mash, Skmer with estimated coverages, and Skmer with true coverages (Skmer*).

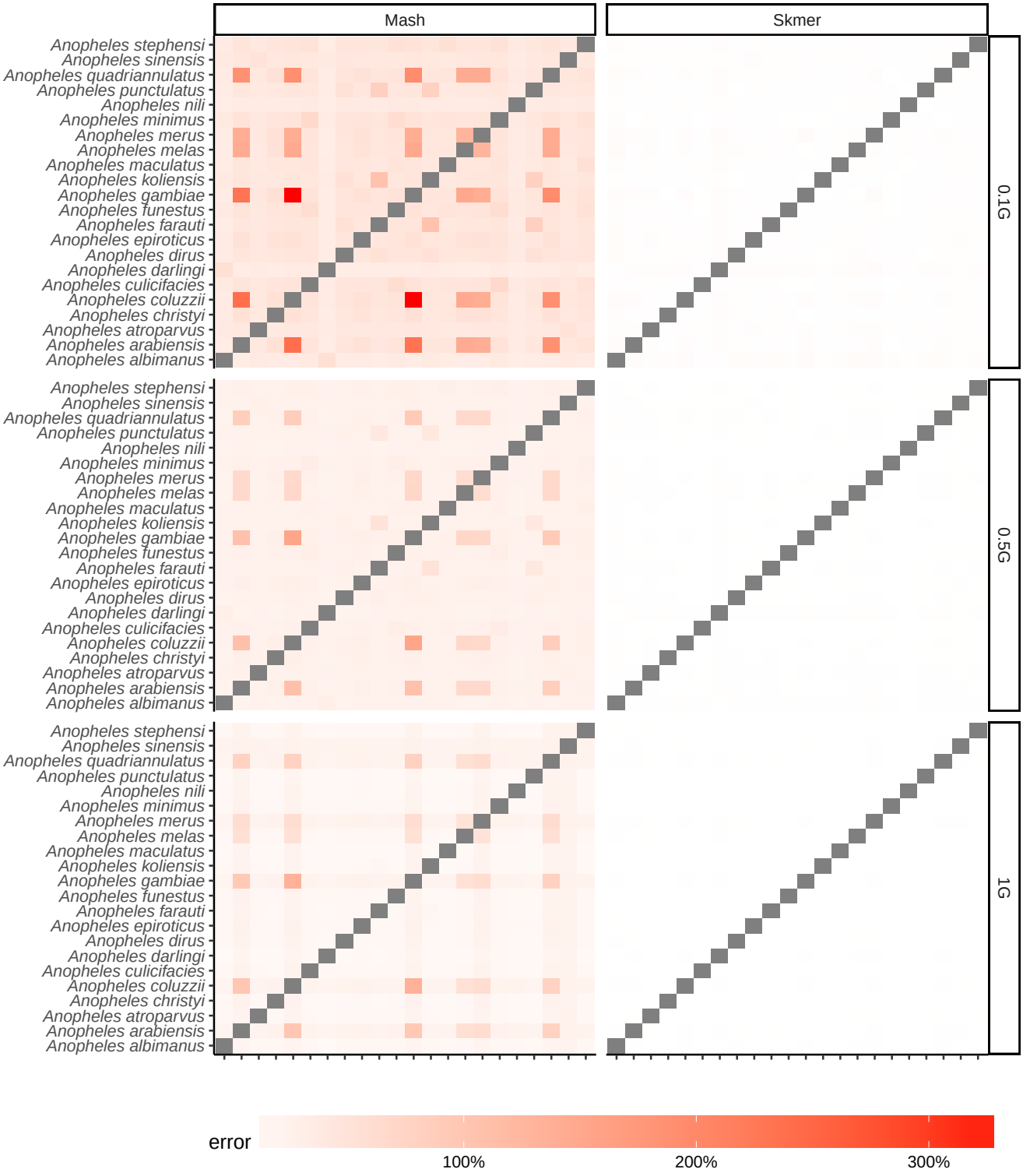


Figure S7: **Comparing the error of Mash and Skmer in distance estimation with fixed amount of sequence from each species.** The dataset of 22 Anopheles genomes, skimmed with 0.1Gb, 0.5Gb, and 1Gb sequence.

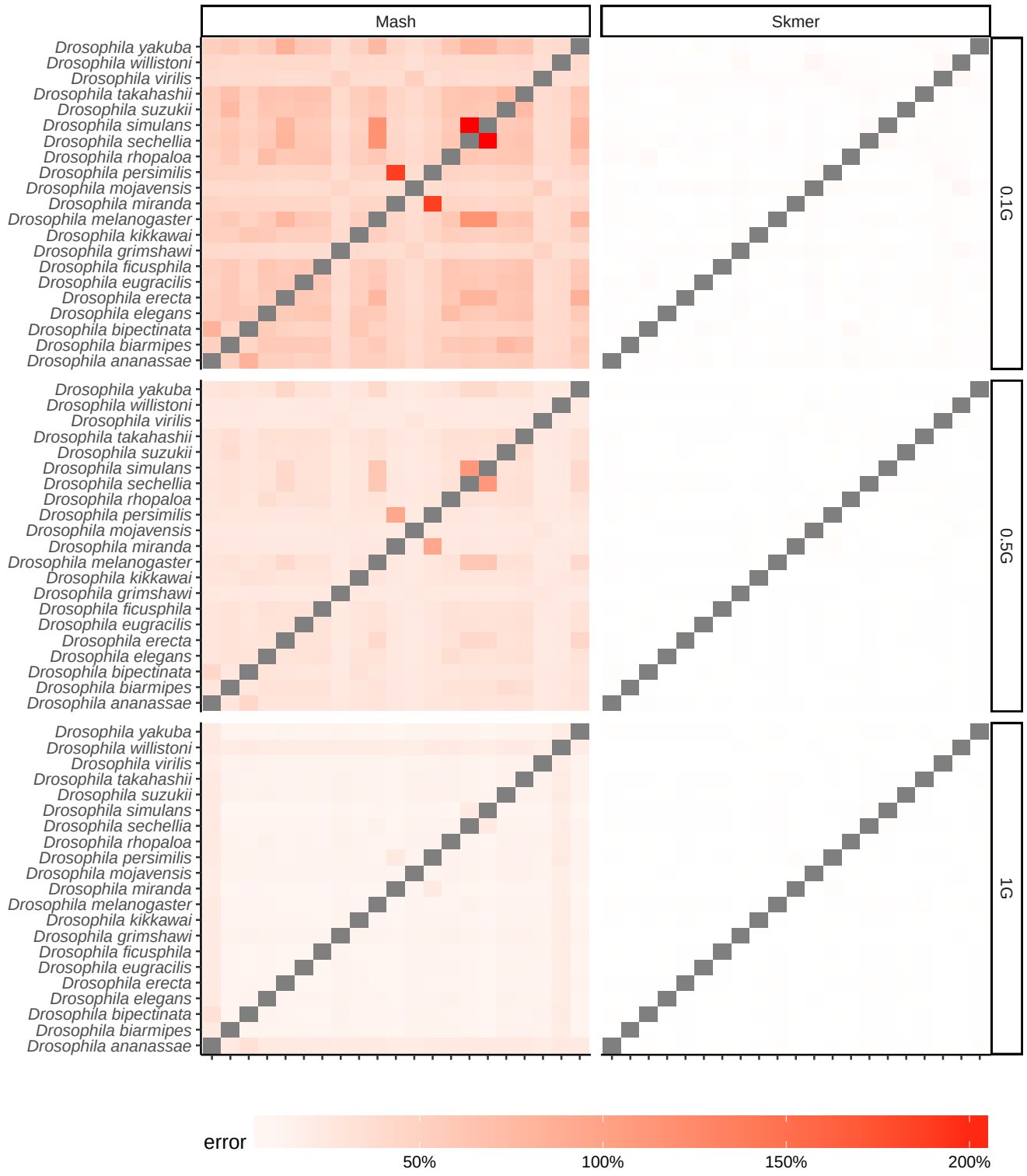


Figure S8: **Comparing the error of Mash and Skmer in distance estimation with fixed amount of sequence from each species.** The dataset of 21 Drosophila genomes, skimmed with 0.1Gb, 0.5Gb, and 1Gb sequence.

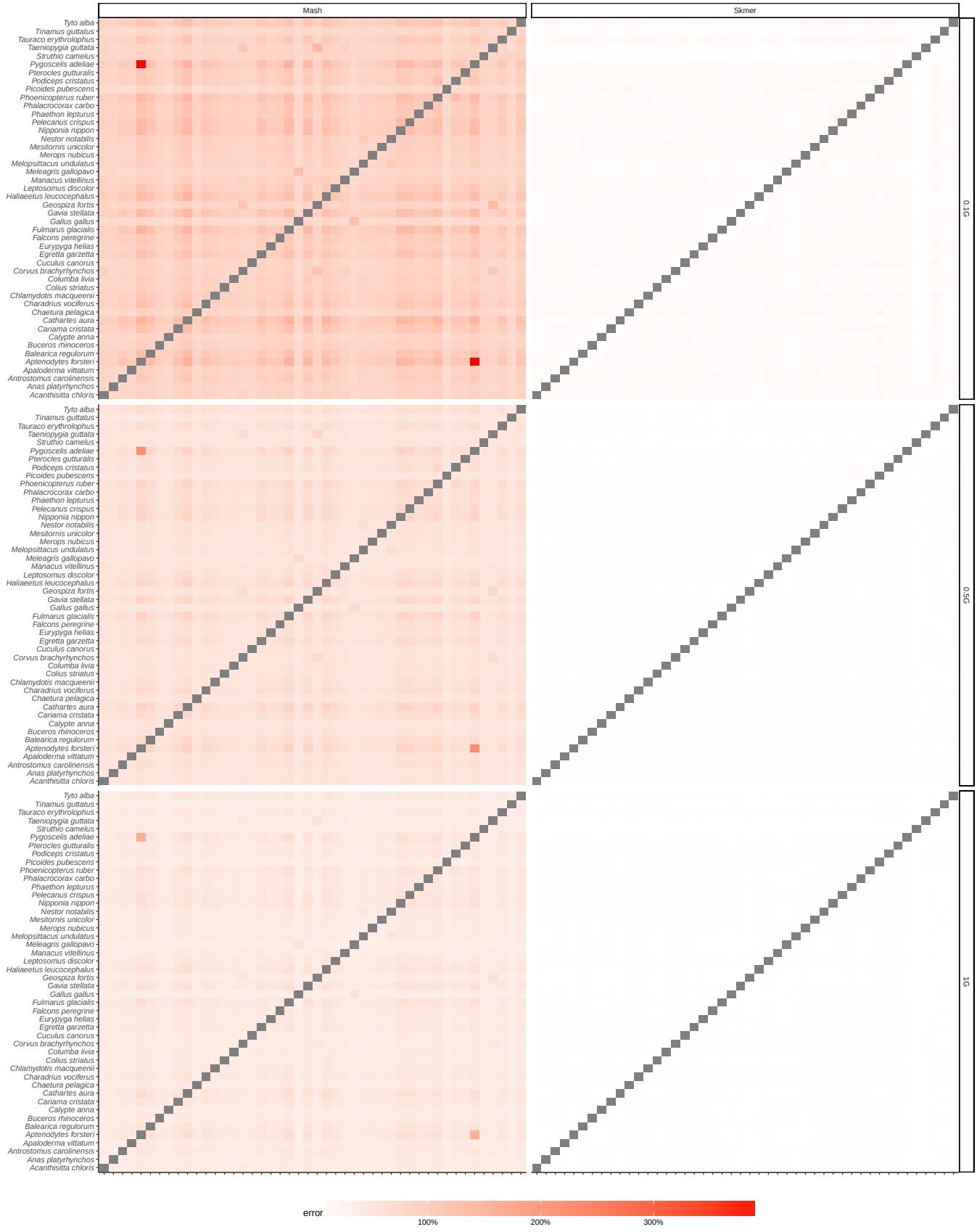


Figure S9: **Comparing the error of Mash and Skmer in distance estimation with fixed amount of sequence from each species.** The dataset of 47 avian genomes, skimmed with 0.1Gb, 0.5Gb, and 1Gb sequence.

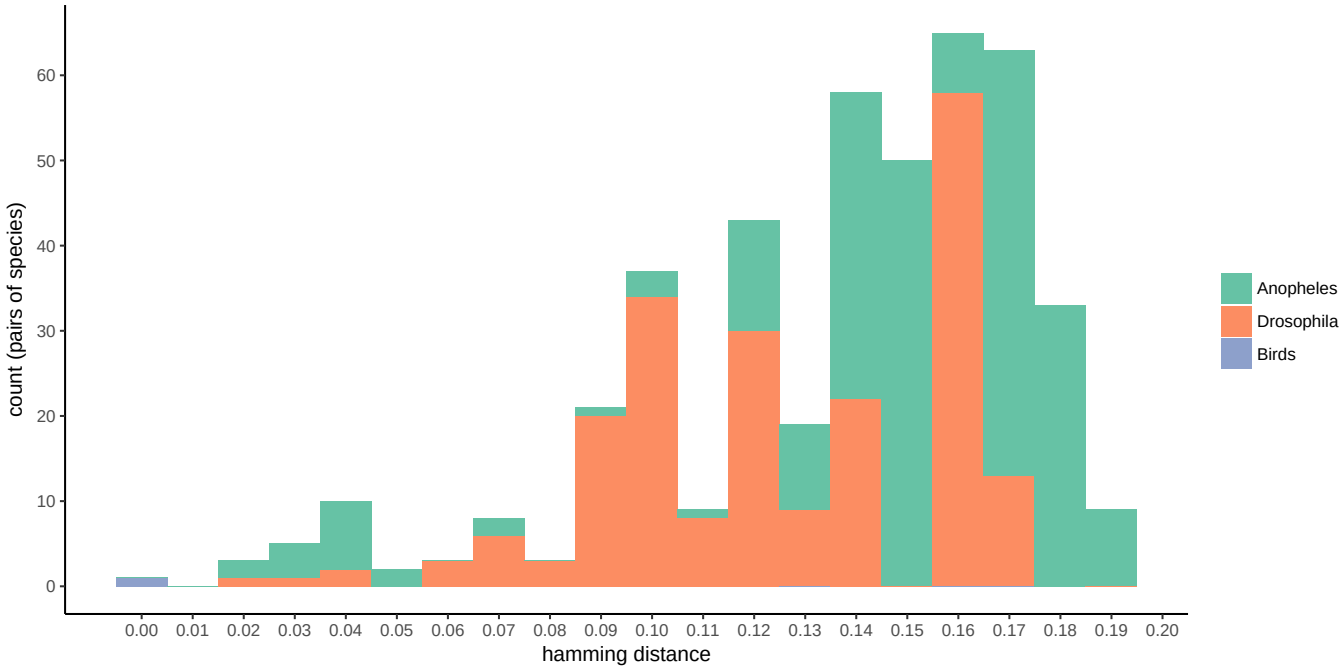


Figure S10: **The histogram of hamming distances between species from the same genus among the Anopheles, Drosophila, and birds datasets.** Distances computed based on full assemblies. The only species from the same genus with hamming distance less than 0.01 were the two eagle species (*H. albicilla* and *H. leucocephalus*).

Table S3: GenBank accession numbers and URLs for Anopheles genomes

Species	GenBank assembly accession	URL
Anopheles albimanus	GCA_000349125.1	http://www.insect-genome.com/data/genome_download/Anopheles_albimanus/Anopheles_albimanus_genomic.fasta.gz
Anopheles arabiensis	GCA_000349185.1	http://www.insect-genome.com/data/genome_download/Anopheles_arabiensis/Anopheles_arabiensis_genomic.fasta.gz
Anopheles atroparvus	GCA_000473505.1	http://www.insect-genome.com/data/genome_download/Anopheles_atroparvus/Anopheles_atroparvus_genomic.fasta.gz
Anopheles christyi	GCA_000349165.1	http://www.insect-genome.com/data/genome_download/Anopheles_christyi/Anopheles_christyi_genomic.fasta.gz
Anopheles coluzzii	-	http://www.insect-genome.com/data/genome_download/Anopheles_coluzzii/Anopheles_coluzzii_genomic.fasta.gz
Anopheles culicifacies	GCA_000473375.1	http://www.insect-genome.com/data/genome_download/Anopheles_culicifacies/Anopheles_culicifacies_genomic.fasta.gz
Anopheles darlingi	GCA_000211455.3	http://www.insect-genome.com/data/genome_download/Anopheles_darlingi/Anopheles_darlingi_genomic.fasta.gz
Anopheles dirus	GCA_000349145.1	http://www.insect-genome.com/data/genome_download/Anopheles_dirus/Anopheles_dirus_genomic.fasta.gz
Anopheles epiroticus	GCA_000349105.1	http://www.insect-genome.com/data/genome_download/Anopheles_epiroticus/Anopheles_epiroticus_genomic.fasta.gz
Anopheles farauti	GCA_000956265.1	http://www.insect-genome.com/data/genome_download/Anopheles_farauti/Anopheles_farauti_genomic.fasta.gz
Anopheles funestus	GCA_000349085.1	http://www.insect-genome.com/data/genome_download/Anopheles_funestus/Anopheles_funestus_genomic.fasta.gz
Anopheles gambiae	GCA_000150785.1	http://www.insect-genome.com/data/genome_download/Anopheles_gambiae/Anopheles_gambiae_genomic.fasta.gz
Anopheles koliensis	GCA_000956275.1	http://www.insect-genome.com/data/genome_download/Anopheles_koliensis/Anopheles_koliensis_genomic.fasta.gz
Anopheles maculatus	GCA_000473185.1	http://www.insect-genome.com/data/genome_download/Anopheles_maculatus/Anopheles_maculatus_genomic.fasta.gz
Anopheles melas	GCA_000473525.2	http://www.insect-genome.com/data/genome_download/Anopheles_melas/Anopheles_melas_genomic.fasta.gz
Anopheles merus	GCA_000473845.2	http://www.insect-genome.com/data/genome_download/Anopheles_merus/Anopheles_merus_genomic.fasta.gz
Anopheles minimus	GCA_000349025.1	http://www.insect-genome.com/data/genome_download/Anopheles_minimus/Anopheles_minimus_genomic.fasta.gz
Anopheles nili	GCA_000439205.1	http://www.insect-genome.com/data/genome_download/Anopheles_nili/Anopheles_nili_genomic.fasta.gz
Anopheles punctulatus	GCA_000956255.1	http://www.insect-genome.com/data/genome_download/Anopheles_punctulatus/Anopheles_punctulatus_genomic.fasta.gz
Anopheles quadriannulatus	GCA_000349065.1	http://www.insect-genome.com/data/genome_download/Anopheles_quadriannulatus/Anopheles_quadriannulatus_genomic.fasta.gz
Anopheles sinensis	GCA_000441895.2	http://www.insect-genome.com/data/genome_download/Anopheles_sinensis/Anopheles_sinensis_genomic.fasta.gz
Anopheles stephensi	GCA_000300775.2	http://www.insect-genome.com/data/genome_download/Anopheles_stephensi/Anopheles_stephensi_genomic.fasta.gz

Table S4: GenBank accession numbers and URLs for Drosophila genomes

Species	GenBank assembly accession	URL
Drosophila ananassae	GCA_000005115.1	http://www.insect-genome.com/data/genome_download/Drosophila_ananassae/Drosophila_ananassae_genomic.fasta.gz
Drosophila biarmipes	GCA_000233415.2	http://www.insect-genome.com/data/genome_download/Drosophila_biarmipes/Drosophila_biarmipes_genomic.fasta.gz
Drosophila bipectinata	GCA_000236285.2	http://www.insect-genome.com/data/genome_download/Drosophila_bipectinata/Drosophila_bipectinata_genomic.fasta.gz
Drosophila elegans	GCA_000224195.2	http://www.insect-genome.com/data/genome_download/Drosophila_elegans/Drosophila_elegans_genomic.fasta.gz
Drosophila erecta	GCA_000005135.1	http://www.insect-genome.com/data/genome_download/Drosophila_erecta/Drosophila_erecta_genomic.fasta.gz
Drosophila eugracilis	GCA_000236325.2	http://www.insect-genome.com/data/genome_download/Drosophila_eugracilis/Drosophila_eugracilis_genomic.fasta.gz
Drosophila ficusphila	GCA_000220665.2	http://www.insect-genome.com/data/genome_download/Drosophila_ficusphila/Drosophila_ficusphila_genomic.fasta.gz
Drosophila grimshawi	GCA_000005155.1	http://www.insect-genome.com/data/genome_download/Drosophila_grimshawi/Drosophila_grimshawi_genomic.fasta.gz
Drosophila kikkawai	GCA_000224215.2	http://www.insect-genome.com/data/genome_download/Drosophila_kikkawai/Drosophila_kikkawai_genomic.fasta.gz
Drosophila melanogaster	GCA_000778455.1	http://www.insect-genome.com/data/genome_download/Drosophila_melanogaster/Drosophila_melanogaster_genomic.fasta.gz
Drosophila miranda	GCA_000269505.2	http://www.insect-genome.com/data/genome_download/Drosophila_miranda/Drosophila_miranda_genomic.fasta.gz
Drosophila mojavensis	GCA_000005175.1	http://www.insect-genome.com/data/genome_download/Drosophila_mojavensis/Drosophila_mojavensis_genomic.fasta.gz
Drosophila persimilis	GCA_000005195.1	http://www.insect-genome.com/data/genome_download/Drosophila_persimilis/Drosophila_persimilis_genomic.fasta.gz
Drosophila rhopaloa	GCA_000236305.2	http://www.insect-genome.com/data/genome_download/Drosophila_rhopaloa/Drosophila_rhopaloa_genomic.fasta.gz
Drosophila sechellia	GCA_000005215.1	http://www.insect-genome.com/data/genome_download/Drosophila_sechellia/Drosophila_sechellia_genomic.fasta.gz
Drosophila simulans	GCA_000259055.1	http://www.insect-genome.com/data/genome_download/Drosophila_simulans/Drosophila_simulans_genomic.fasta.gz
Drosophila suzukii	GCA_000472105.1	http://www.insect-genome.com/data/genome_download/Drosophila_suzukii/Drosophila_suzukii_genomic.fasta.gz
Drosophila takahashii	GCA_000224235.2	http://www.insect-genome.com/data/genome_download/Drosophila_takahashii/Drosophila_takahashii_genomic.fasta.gz
Drosophila virilis	GCA_000005245.1	http://www.insect-genome.com/data/genome_download/Drosophila_virilis/Drosophila_virilis_genomic.fasta.gz
Drosophila willistoni	GCA_000005925.1	http://www.insect-genome.com/data/genome_download/Drosophila_willistoni/Drosophila_willistoni_genomic.fasta.gz
Drosophila yakuba	GCA_000005975.1	http://www.insect-genome.com/data/genome_download/Drosophila_yakuba/Drosophila_yakuba_genomic.fasta.gz

Table S5: GenBank accession numbers and URLs for avian genomes

Species	GenBank assembly accession	URL
Acanthisitta chloris	GCA_000695815.1	http://dx.doi.org/10.5524/101015
Anas platyrhynchos	GCA_000355885.1	http://dx.doi.org/10.5524/101001
Antrostomus carolinensis	GCA_000700745.1	http://dx.doi.org/10.5524/101019
Apaloderma vittatum	GCA_000703405.1	http://dx.doi.org/10.5524/101016
Aptenodytes forsteri	GCA_000699145.1	http://dx.doi.org/10.5524/100005
Balearica regulorum	GCA_000709895.1	http://dx.doi.org/10.5524/101017
Buceros rhinoceros	GCA_000710305.1	http://dx.doi.org/10.5524/101018
Calypte anna	GCA_000699085.1	http://dx.doi.org/10.5524/101004
Cariama cristata	GCA_000690535.1	http://dx.doi.org/10.5524/101020
Cathartes aura	GCA_000699945.1	http://dx.doi.org/10.5524/101021
Chaetura pelagica	GCA_000747805.1	http://dx.doi.org/10.5524/101005
Charadrius vociferus	GCA_000708025.2	http://dx.doi.org/10.5524/101007
Chlamydotis macqueenii	GCA_000695195.1	http://dx.doi.org/10.5524/101022
Colius striatus	GCA_000690715.1	http://dx.doi.org/10.5524/101023
Columba livia	GCA_000337935.1	http://dx.doi.org/10.5524/100007
Corvus brachyrhynchos	GCA_000691975.1	http://dx.doi.org/10.5524/101008
Cuculus canorus	GCA_000709325.1	http://dx.doi.org/10.5524/101009
Egretta garzetta	GCA_000687185.1	http://dx.doi.org/10.5524/101002
Eurypyga helias	GCA_000690775.1	http://dx.doi.org/10.5524/101024
Falcons peregrine	GCA_000337955.1	http://dx.doi.org/10.5524/101006
Fulmarus glacialis	GCA_000690835.1	http://dx.doi.org/10.5524/101025
Gallus gallus	GCA_000002315.3	ftp://climb.genomics.cn/pub/10.5524/100001_101000/101000/chicken/
Gavia stellata	GCA_000690875.1	http://dx.doi.org/10.5524/101026
Geospiza fortis	GCA_000277835.1	http://dx.doi.org/10.5524/100040
Haliaeetus albicilla	GCA_000691405.1	http://dx.doi.org/10.5524/101027
Haliaeetus leucocephalus	GCA_000737465.1	http://dx.doi.org/10.5524/101040
Leptosomus discolor	GCA_000691785.1	http://dx.doi.org/10.5524/101028
Manacus vitellinus	GCA_000692015.2	http://dx.doi.org/10.5524/101010
Meleagris gallopavo	GCA_000146605.3	ftp://climb.genomics.cn/pub/10.5524/100001_101000/101000/turkey/
Melopsittacus undulatus	GCA_000238935.1	http://dx.doi.org/10.5524/100059
Merops nubicus	GCA_000691845.1	http://dx.doi.org/10.5524/101029
Mesitornis unicolor	GCA_000695765.1	http://dx.doi.org/10.5524/101030
Nestor notabilis	GCA_000696875.1	http://dx.doi.org/10.5524/101031
Nipponia nippon	GCA_000708225.1	http://dx.doi.org/10.5524/101003
Pelecanus crispus	GCA_000687375.1	http://dx.doi.org/10.5524/101032
Phaethon lepturus	GCA_000687285.1	http://dx.doi.org/10.5524/101033
Phalacrocorax carbo	GCA_000708925.1	http://dx.doi.org/10.5524/101034
Phoenicopterus ruber	GCA_000687265.1	http://dx.doi.org/10.5524/101035
Picoides pubescens	GCA_000699005.1	http://dx.doi.org/10.5524/101012
Podiceps cristatus	GCA_000699545.1	http://dx.doi.org/10.5524/101036
Pterocles gutturalis	GCA_000699245.1	http://dx.doi.org/10.5524/101037
Pygoscelis adeliae	GCA_000699105.1	http://dx.doi.org/10.5524/100006
Struthio camelus	GCA_000698965.1	http://dx.doi.org/10.5524/101013
Taeniopygia guttata	GCA_000151805.2	ftp://climb.genomics.cn/pub/10.5524/100001_101000/101000/zebrafinch/
Tauraco erythrolophus	GCA_000709365.1	http://dx.doi.org/10.5524/101038
Tinamus guttatus	GCA_000705375.2	http://dx.doi.org/10.5524/101014
Tyto alba	GCA_000687205.1	http://dx.doi.org/10.5524/101039

Table S6: **The average error of Mash and Skmer with different estimates of ϵ, for two models of sequencing error.** The Anopheles dataset; Each species skimmed with 0.5Gb sequence.

Method	Constant error rate ($\epsilon = 0.01$)	HiSeq2000 error profile
Mash	28.51% (0.74%)	25.01% (0.60%)
Skmer (0.001)	5.38% (0.15%)	5.53% (0.15%)
Skmer (0.005)	4.12% (0.14%)	2.10% (0.09%)
Skmer (0.01)	0.82% (0.02%)	3.39% (0.11%)
Skmer (0.02)	10.32% (0.40%)	12.68% (0.53%)

* The standard error of the mean is provided in parentheses.