Variance in Variants: Propagating Genome Sequence Uncertainty into Phylogenetic Lineage Assignment

David Champredon*, Devan Becker*, Connor Chato, Gopi Gugan, Art Poon

*contributed equally

Contents

Abstract

Genetic sequencing is subject to many different types of errors, but most analyses treat the resultant sequences as if they are known without error. Next generation sequencing methods rely on significantly larger numbers of reads in exchange for a loss of accuracy in each individual read. Still, the coverage of such machines is imperfect and leaves uncertainty in many of the base calls. On top of this machine-level uncertainty, there is uncertainty induced by human error. In this work, we demonstrate that the uncertainty in sequencing techniques will affect downstream analysis and propose a straightforward method to propagate the uncertainty.

Our method uses a probabilistic matrix representation of individual sequences which incorporates base quality scores as a measure of uncertainty that naturally lead to resampling and replication as a framework for uncertainty propagation. With the matrix representation, resampling possible base calls according to quality scores provides a bootstrap- or prior distribution-like first step towards genetic analysis. Analyses based on these re-sampled sequences will include an more complete evaluation of the error involved in such analyses.

We demonstrate our resampling method on SARS-CoV-2 data. The resampling procedures adds a linear computational cost to the analyses, but the large impact on

the variance in downstream estimates makes it clear that ignoring this uncertainty may lead to overly confident conclusions. We show that SARS-CoV-2 lineage designations via Pangolin are much less certain than the bootstrap support reported by Pangolin would imply and the clock rate estimates for SARS-CoV-2 are much more variable than reported.

1   Introduction

Generating a genetic sequence from a biological sample is a complex process. Nucleic acids must be extracted from the sample while avoiding contamination by foreign material. If working with RNA, then we must use a reverse transcriptase reaction to convert the RNA into DNA. PCR amplification is often employed to enrich the sample for the target of interest. For next-generation sequencing (NGS) protocols, we have to generate a sequencing library for instance by random shearing of nucleic acids into fragments that are ligated onto special 'adaptors'. TODO: ref? NGS procedures such as sequencing by synthesis suffer from greater error rate relative to conventional Sanger dye-terminator sequencing, although these rates have continued to improve with new technologies (Fuller et al., 2009). In addition, the short reads produced by NGS platforms need to be aligned — either by alignment against a reference genome, *de novo* assembly, or a combination of the two — to reconstruct a consensus sequence using one or more bioinformatic programs. Errors can be introduced in any one of these steps (Beerenwinkel and Zagordi, 2011; O'Rawe et al., 2015).

In some cases, naturally occurring variation, *i.e.*, genetic polymorphisms, or variation induced by experimental error is directly quantified and encoded into the output. For example, mixed peaks in sequence chromatograms produced from dye terminator sequencing by capillary electrophoresis are assigned standard IUPAC codes (*e.g.,* Y for C or T) when the base calling program cannot determine which base is dominant (NC-IUB, 1986). Ewing and Green (1998) and Richterich (1998) both showed that estimates of the base call quality, typically quantified as Phred quality scores ($Q = -10 \log_{10} P$, where P is the estimated error probability), can be an accurate estimate of the number of errors that the machines at the time would make. The accuracy of Phred quality scores has been disputed and other methods for quantifying error probabilities have been proposed (Li et al., 2004; DePristo et al., 2011; Li et al., 2009b), Phred scores remain the standard means of reporting the estimated error probabilities for current sequencing platforms. Generally, these scores are used to censor the base calls (*i.e.*, label them "N" rather than A, T, C or G) if the estimated probability of error exceeds a predefined threshold. It is also common practice to remove the sequence from further analysis if the total number of censored bases exceeds a maximum tolerance; *e.g.,* Doronina (2005); Robasky et al. (2014); O'Rawe et al. (2015). Some authors/tools, such as Lippi et al. (2020), use a more sophisticated form of quality control, but still use the resultant reads to form a consensus sequence with no measure of uncertainty. Furthermore, some studies have extended the concept of per-base error probabilities to calculate the joint likelihoods of partial or full sequences. For example, DePristo et al. (2011) and Gompert and Buerkle (2011) incorporate adjusted Phred scores into a likelihood framework to generate more accurate estimates of genetic diversity within a population; this approach has subsequently been used to develop new estimators of genetic diversity (Fumagalli et al., 2013). Kuo et al. (2018) recently used a similar approach to develop a statistical test of whether a given genome sequence is consistent with a specified alternative sequence. In general, the reported error probabilities from NGS technologies are primarily used for filtering low quality sequences and improving alignment algorithms (which both result in a consensus sequence that is treated as error-free) or for hypothesis tests concerning small collections (usually pairs) of sequences.

The uncertainty present in the sequences are seldom propagated to downstream analyses. For example, methods for sequence alignment and homology searches generally employ

heuristic algorithms that utilize similarity scores that do not explicitly incorporate the probabilities of sequencing errors. Moreover, methods to reconstruct the evolutionary relationships among sequences as a phylogenetic tree tend to interpret ambiguous base calls as completely missing data, although some exceptions are found in the literature, *e.g.,* DePristo et al. (2011). This problem is exacerbated when each sequence represents the consensus of diverse copies of a genome, such as rapidly evolving virus populations where genuine polymorphisms are confounded with sequencing error. See Schneider (2002) for more criticisms of the use of consensus sequences, along with visualizations (Schneider and Stephens, 1990, called *sequence logos*,) to display the deviations from a consensus.

Though rare, some studies have proposed methods for propogation of uncertainty from one step to later steps of an analysis. O'Rawe et al. (2015) suggest methods for propagation of sequence-level uncertainty into determining whether two subjects have the same alleles, as well as estimating confidence intervals for allele frequencies. Another exception can be found in Kuhner and McGill (2014), who incorporate an assumed or estimated error rate for the entire sequence into the calculation of a phylogenetic tree and found that incorporation of errors makes the inferred branch lengths much closer to the true (simulated) branch lengths. Though they did not use nucleotide-level uncertainty, Gompert and Buerkle (2011) incorporate the coverage of NGS technologies as part of the uncertainty of estimates for the frequency of allelles in a population. Clement et al. (2010) present an alignment algorithm (called GNUMAP) that takes nucleotide-level uncertainty into account. Their method incorporates Position Weight Matrices into a method of scoring multiple possible matches against a reference genome in order to choose the best alignment. These studies are the exceptions, rather than the rules, and their methods have not yet attained widespread use.

We present a simple general-purpose framework that can be incorporated into any analysis of genetic sequence data. This framework involves converting the uncertainty scores into a matrix of probabilities, and repeatedly sampling from this matrix and using the resultant samples in downstream analysis. Unlike likelihood-based approaches, we do not make assumptions about the underlying patterns or distributions in the data. In so doing, we can gain more accurate estimation of the errors at the expense of computation time. Our technique is amenable to quality score adjustments prior to applying our methods. We demonstrate the impact of propagating sequence uncertainty by applying our methods to the problem of classifying SARS-CoV-2 genomes into predefined clusters known as 'lineages', several of which correspond to variants carrying mutations that are known to confer an advantage to virus transmission or infectivity. We also analyse a collection of SARS-CoV-2 sequences to demonstrate that the clock rate from a root-to-tip regression is much more variable than studies relying on deterministic sequences would conclude.

2   Methods

2.1   Probabilistic representation of sequences

Here, we describe two theoretical frameworks to model sequence uncertainty at the *nucleotide level* or at the *sequence level*. In both frameworks, the sequence of nucleotides from a biological sample is not treated as a single unambiguous observation (known without error), but rather as a collection of possible sequences weighted by their probability.

2.1.1   Nucleotide-level uncertainty

To represent the uncertainty at each position along the genome we introduce the following matrix, which we will refer to as a probabilistic sequence and denote $\mathcal{S}$:

$$\mathcal{S} = \begin{array}{c} \\ \texttt{A} \\ \texttt{C} \\ \texttt{G} \\ \texttt{T} \\ \texttt{-} \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & \ldots & \ell \end{array} \\ \begin{pmatrix} \mathcal{S}_{A,1} & \mathcal{S}_{A,2} & \ldots & \mathcal{S}_{A,\ell} \\ \mathcal{S}_{C,1} & \mathcal{S}_{C,2} & \ldots & \mathcal{S}_{C,\ell} \\ \mathcal{S}_{G,1} & \mathcal{S}_{G,2} & \ldots & \mathcal{S}_{G,\ell} \\ \mathcal{S}_{T,1} & \mathcal{S}_{T,2} & \ldots & \mathcal{S}_{T,\ell} \\ \mathcal{S}_{x,1} & \mathcal{S}_{x,2} & \ldots & \mathcal{S}_{x,\ell} \end{pmatrix} \end{array} \quad (1)$$

Each column represents a position in a nucleotide sequence of length ℓ. Each row represents one of the four nucleotides $\texttt{A,C,G,T}$, as well as an empty position "$\texttt{-}$" that symbolizes a recorded deletion rather than missing data. Hence, $\mathcal{S}$ is a $5 \times \ell$ matrix.

The elements of the probability sequence represent the probability that a nucleotide exists at a given position:

$$\mathcal{S}_{\texttt{n},j} = \mathbb{P}(\text{nucleotide } \texttt{n} \text{ is at position } j) \quad (2)$$

with the special case for a deletion:

$$\mathcal{S}_{\texttt{-},j} = \mathbb{P}(\text{empty position } j). \quad (3)$$

Note that we have for all $1 \le j \le \ell$:

$$\sum_n \mathcal{S}_{n,j} = 1 \quad (4)$$

Also, the sequence length is stochastic if $0 < \mathcal{S}_{-,i} < 1$ for at least one i. The probability that the sequence has the maximum length ℓ is $\prod_{i=1}^{\ell}(1-\mathcal{S}_{-,i})$. The nucleotide (or deletion) drawn at each position is independent from all the others, so there are up to 5^ℓ possible different sequences for a given probabilistic nucleotide sequence, but these sequences are *not* equally probable.

A major limitation of this probabilistic representation of a sequence is that we lose all information on linkage disequilibrium. This is especially problematic for recording insertions. Instead, we assume that every nucleotide is an independent observation. For example, a probability sequence populated from short read data from a diverse population would not store the information that two polymorphisms were always observed in the same reads, *i.e.*, in complete linkage disequilibrium. We also lose information about autocorrelation in sequencing error, such as clusters of miscalled bases associated with later cycles of sequencing-by-synthesis platforms. Sequence chromatograms are affected by the same loss of information.

We note that this representation is similar to the "CATG" file type as described in Kozlov (2018), which indicates the likelihoods of each nucleotide in an aligned mapping for multiple taxa. This file type is able to be used by RAxML-NG to estimate an overall error rate which is then used to estimate phylogenetic trees. Our probability sequence is also similar in concept to Position Weight Matrices (PWMs, Stormo et al., 1982) which are built according to the frequency of each base at each position of a multiple alignment. Our construction differs in that we are creating one matrix per sequence where the entries are weighted according to error probability within that sequence, rather than one matrix for a collection of sequences. However, methods that accept PWMs will be applicable to our probability sequences (and *vice-versa*).

2.1.2 Sequence-level uncertainty

A significant problem of storing probabilities at the level of individual nucleotides is that generating a sequence from this matrix requires drawing ℓ independent outcomes. For

example, the reference SARS-CoV-2 genome is 29,903 nucleotides, and a substantial number of naturally-occurring sequence insertions have been described. Thus it would not be surprising if ℓ exceeded 30,000 nucleotides (nt). The majority of these technically possible 5^ℓ sequences are not biologically plausible. Therefore, we formulate an ordered subset $\mathcal{B} = (\mathcal{B}_i)_{i \in \{1...m\}}$ of the first m most likely sequences, which are ranked in descending order by the joint probability of nucleotide composition. Note that the $\mathcal{B}_i$ do not necessarily have the same length. The observed genetic sequence, s^*, is a sample from a specified discrete probability distribution a:

$$\mathbb{P}(s^* = \mathcal{B}_i | i \leq m) = a(i) \tag{5}$$

This compact and approximate representation drastically reduces the number of operations to one sample, after some pre-processing to calcualte a. The observed consensus sequence s^* is guaranteed to be a member of $\mathcal{B}$ if $\mathcal{S}_{s(j),j} > 0.5 \; \forall \; j$ where $s(j)$ is the j-th nucleotide of s^*; indeed, it is guaranteed to the highest ranked member $i = 0$. We refer to any member of the set $\mathcal{B}$ as a *sequence-level probabilistic sequence*. Note that because a is a probability distribution, we must have $\sum_{i=1}^{m} a(i) = 1$. In other words, this probability is conditional on the sequence being in $\mathcal{B}$.

For example, suppose that we have the following nucleotide-level probabilistic sequence:

$$
\mathcal{S} =
\begin{array}{c}
\texttt{A} \\
\texttt{C} \\
\texttt{G} \\
\texttt{T} \\
\texttt{--}
\end{array}
\begin{pmatrix}
\overset{1}{0.9} & \overset{2}{0.05} & \overset{3}{0.99} & \overset{4}{0} & \overset{5}{0} & \overset{6}{0.6} \\
0 & 0.8 & 0 & 0 & 0.1 & 0.1 \\
0.1 & 0.15 & 0 & 0.3 & 0.9 & 0 \\
0 & 0 & 0.01 & 0.7 & 0 & 0.3 \\
0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
$$

such that there are $2 \times 3 \times 2^3 \times 3 = 144$ possible sequences. The most likely sequence has the highest joint nucleotide probability: `ACATGA` with probability 0.2694 ($0.9 \times 0.8 \times 0.99 \times 0.7 \times 0.9 \times 0.6$). If there is a positive probability of deletion for at least one position, then the sequence has a variable length. Large genomes or sequencing targets will result in vanishingly small probabilities for all sequences, and thus calculations on the log scale may be necessary to reduce the chance of numerical underflow.

2.1.3 Deletions and insertions

By construction, the nucleotide-level probabilistic sequence must be defined with its longest possible length. Deletions are naturally modelled with our representation but insertions have to be modelled using deletion probabilities.

$$
\mathcal{S} =
\begin{array}{c}
\texttt{A} \\
\texttt{C} \\
\texttt{G} \\
\texttt{T} \\
\texttt{--}
\end{array}
\begin{pmatrix}
\overset{1}{0} & \overset{2}{0} & \overset{3}{1} & \overset{4}{0} & \overset{5}{1} & \overset{6}{0} \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.99 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.01 & 0 & 1 \\
0 & 0.01 & 0 & 0.99 & 0 & 0
\end{pmatrix}
\tag{6}
$$

The low deletion probability for position 2 is straightforward to interpret: about 1% of the time, nucleotide `G` at position 2 is deleted. The high deletion probability for position 4 means there is a 1% chance of a `T` insertion at this position (Table 1).

sequence	frequency
`CGAAT`	common, 98% of the time
`CAAT`	rare (1% frequency) `G` deletion at position 2,
`CGATAT`	rare (1% frequency) `T` insertion at position 4
`CATAT`	very rare (0.01% frequency) deletion and insertion

Table 1: Representation of insertions and deletions from $\mathcal{S}$ defined in Equation (6)

203 The representation of deletions and insertions with a sequence-level probabilistic sequence
204 (not nucleotide-level) is straightforward, because in this framework the sequences are
205 explicitly written out — so are their deletions/insertions.

2.2 Constructing the probability sequence

2.2.1 SAM files

208 In most next-generation sequencing applications, the estimated probability of sequenc-
209 ing error is quantified with the quality (or "Phred") score attributed to each base call
210 produced by sequencing instrument. The quality score Q is directly related to this error
211 probability: $\epsilon = 10^{-Q/10}$ (Ewing and Green, 1998), where Q typically ranges between 1
212 and 60 (with 60 being the lowest probability of error), depending on the sequencing plat-
213 form and version of base-calling software. It is important to note that this quality score
214 only measures the probability of error from the machine; $1 - \epsilon$ should not be interpreted
215 as the probability that a call is correct.

216 More formally, the probability that the base call is correct is expressed as:

$$\mathbb{P}(\text{nucleotide} = X \mid \text{observed nucleotide} = X) = 1 - \epsilon \qquad (7)$$

217 Unfortunately, quality scores have no information on the probabilities of the three other
218 possible nucleotides if the base call is incorrect. In the absence of information about the
219 other bases, we assume that these other probabilities are uniformly distributed.

220 Raw short read data are typically recorded in a FASTQ format that stores both the
221 sequences (base calls) and base-specific quality scores. Since the reads often correspond
222 to different positions of the target nucleic acid, *e.g.,* randomly sheared genomic DNA,
223 it is necessary to align the reads to identify base calls on different reads that represent
224 the same genome position. This alignment step can be accomplished by mapping reads
225 to a reference genome, by the *de novo* assembly of reads, or a hybrid approach that
226 incorporates both methods. The aligned outputs are frequently recorded in the tabular
227 Sequence Alignment/Map (SAM) format (Li et al., 2009a). Each row represents a short
228 read, including the raw nucleotide sequence and quality strings; the optimal placement
229 of the read with respect to the reference sequence (as an integer offset); and the compact
230 idiosyncratic gapped alignment report (CIGAR), an application-specific serialization of
231 the edit operations required to align the read to the reference.

232 We employed the following procedure to construct the nucleotide-level probabilistic se-
233 quence from the contents of a SAM file. We initialize aligned sequence and quality strings
234 with '-' and '!' ($Q = 0$) repeated $r - 1$ times, respectively, where r is the (one-indexed)
235 position of the leftmost nucleotide in the reference genome. Next, we tokenize the CIGAR
236 string into length-operation tuples, which determine how bases and quality scores from
237 the raw strings are appended to the aligned versions. Deleted bases ('D' operations) are
238 not assigned Phred scores, so we assume them to have 0 error probability.

239 Insertions ('I' operations) are non-trivial to include in the probabilistic sequence. Consider
240 a short read with two bases inserted at position j (say, an A at position $j + 1$ and a T at
241 position $j + 2$) and a short read with one insertion at position j (say, a C). It is entirely
242 ambiguous whether the single insertion (C) aligns with the first insertion (A) or the second
243 insertion (T) of the first short read. For our purposes, we only consider the alignment of
244 these sequences with a reference sequence and thus do not consider insertions.

245 Some NGS platforms use paired-end reads where the same nucleic acid template is read
246 in both directions. In these situations, we simply adjust all values by a factor of two.
247 For bases where the paired-end reads overlap, this has the effect of averaging the base
248 probability $1 - \epsilon$. For example, if these values are 90% for A in one read and 95% A in its
249 mate, then 0.925 is added to the A row in $\mathcal{S}'$. If the two reads were 60% A and 55% C at
250 the same position, then we would increment the corresponding column vector by (0.375,

$_{251}$ 0.341$\bar{6}$, 0.141$\bar{6}$, 0.141$\bar{6}$). Bases outside of the overlapping region contribute a maximum
$_{252}$ of 0.5 to $\mathcal{S}'$, because the base call on the other read is missing data. This approach has
$_{253}$ the advantage of making the parsing of SAM files trivially parallelizable since we do not
$_{254}$ need to know how reads are paired. In addition, the coverage calculated from $\mathcal{S}'$ is scaled
$_{255}$ to the number of templates rather than the number of reads.

2.2.2   Consensus sequence FASTQ files

$_{257}$ Full length or partial genome sequences are now frequently the product of next-generation
$_{258}$ sequencing, by taking the consensus of the aligned or assembled read data. However,
$_{259}$ the original read data are often not published alongside the consensus sequence. Some
$_{260}$ consensus sequences are released in a format where the bases are annotated with quality
$_{261}$ scores, *e.g.,* FASTQ. There are several programs that provide methods to convert a SAM
$_{262}$ file into a consensus FASTQ file (Li et al., 2004; Keith et al., 2002; Li et al., 2008).
$_{263}$ These programs use slightly different methods for generated consensus quality scores, but
$_{264}$ generally filter quality scores for the majority base. For example, suppose there are three
$_{265}$ reads with the following base calls at position j: A with $Q = 30$, A with $Q = 31$, and C
$_{266}$ with $Q = 15$. Calculation of the consensus quality score will thereby exclude the $Q = 15$
$_{267}$ value.

$_{268}$ This omission makes it challenging for us to generate an $\mathcal{S}$ matrix from a consensus
$_{269}$ FASTQ file. Given the consensus base and its associated quality score at position j, we
$_{270}$ must assume that the other bases are all equally likely with probability $\epsilon_j/3$ (similar to
$_{271}$ Kuo et al. (2018) and Chapter 5 of Kozlov (2018)). For example, let's assume the output
$_{272}$ sequence after fragment sequencing and alignment is ACATG and its associated quality
$_{273}$ scores are respectively $Q = 60, 30, 50, 10, 40$. The probabilistic sequence is:

$$\mathcal{S} = \begin{array}{c} \text{A} \\ \text{C} \\ \text{G} \\ \text{T} \\ \text{–} \end{array} \begin{pmatrix} \overset{1}{1 - 10^{-6}} & \overset{2}{10^{-3}/3} & \overset{3}{1 - 10^{-5}} & \overset{4}{10^{-1}/3} & \overset{5}{10^{-4}/3} \\ 10^{-6}/3 & 1 - 10^{-3} & 10^{-5}/3 & 10^{-1}/3 & 10^{-4}/3 \\ 10^{-6}/3 & 10^{-3}/3 & 10^{-5}/3 & 10^{-1}/3 & 1 - 10^{-4} \\ 10^{-6}/3 & 10^{-3}/3 & 10^{-5}/3 & 1 - 10^{-1} & 10^{-4}/3 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{8}$$

$_{274}$ Usually, the genetic sequence ACATG would be considered as certain and quality scores
$_{275}$ discarded. In contrast, the probability of the sequence ACATG is only 0.899 within the
$_{276}$ probabilistic sequence framework.

$_{277}$ Incorporating deletions in the absence of raw data is also challenging. If one is willing to
$_{278}$ assume global deletion rate, then it is possible to extend the parameterization of $\mathcal{S}$. For
$_{279}$ example, if the probability of a single nucleotide deletion is d, then the probability of the
$_{280}$ called base is $(1 - d_j)(1 - \epsilon_j)$ and the other three nucleotides have probability $(1 - d)\epsilon_j/3$.
$_{281}$ Hence, if we assume the base call is A, the column of the nucleotide-level probabilistic
$_{282}$ sequence for that position is

$$\mathcal{S}(, j) = \begin{array}{c} \text{A} \\ \text{C} \\ \text{G} \\ \text{T} \\ \text{–} \end{array} \begin{pmatrix} \overset{j}{(1 - d)(1 - \epsilon_j)} \\ (1 - d)\epsilon_j/3 \\ (1 - d)\epsilon_j/3 \\ (1 - d)\epsilon_j/3 \\ d \end{pmatrix} \tag{9}$$

$_{283}$ Since the FASTQ file only has a single sequence, we do have the same issues with alignment
$_{284}$ of differing lengths of insertions. In fact, insertions are only insertions relative to the
$_{285}$ reference sequence; they can simply be treated as observed nucleotides with an associated
$_{286}$ quality score. It would be possible to give insertions special treatment, however, by
$_{287}$ defining a global insertion rate. This insertion rate can be expressed as a deletion rate

288 relative to the observed sequence, and thus one minus the insertion rate can be treated
289 as the deletion rate in the probabilistic sequence. As with the deletion rate, this requires
290 an assumption about a global rate which may be arbitrary.

2.2.3   Consensus sequence FASTA files
291

If we do not have access to any base quality information, *e.g.,* the consensus sequence
is published as a FASTA file, then our ability to populate $\mathcal{S}$ is severely limited. Any
uncertainty that we impose upon the data will be a principled assumption. The error
probability at the j position of the consensus sequence can be simulated as a beta distri-
bution, *i.e.,*

$$\epsilon_j \sim \mathrm{Beta}(\alpha, \beta)$$

292 The called base at position j has probability $1 - \epsilon$, and the remaining bases are assigned
293 $\epsilon_j/3$. To incorporate deletions, another probability d can be generated as the *gap proba-*
294 *bility.* With these defined, the nucleotide-level probabilistic sequence at the jth column
295 (assuming the base call at position j was A) can be written as above. This probabilistic
296 sequence is completely fabricated, *i.e.,* not based on any empirical data. However, the
297 sensitivity of an analysis can be evaluated by choosing different values of α, β, and d (*e.g.,*
298 based on previous studies) and propagating these uncertainties into downstream analy-
299 ses. The results from such an analysis would not indicate anything about the sequence
300 itself but could be used to determine how robust the methods are to increased sequence
301 uncertainty.

2.3   Propagation of uncertainty via resampling
302

303 The most general way to propagate uncertainty is through resampling. Given $\mathcal{S}$ and
304 assuming that individual nucleotides are independent outcomes — or precomputing a
305 reduced set of m sequences and calculating the distribution a of their joint probabilities
306 — we can propagate uncertainy by running downstream analyses on each set of sampled
307 sequences.

308 At a nucleotide level, we are sampling from a multinomial distribution. If the jth column
309 of $\mathcal{S}$ is (0.5, 0.2, 0.2, 0.09, 0.01), then we could sample A with 50% probability, C with 20%,
310 etc. As with other sequence analyses, we can censor the positions that do not have enough
311 coverage. We arbitrarily chose to censor any position that had fewer than 10 reads, which
312 can be determined by summing the column of the probabilistic sequence.

313 In a maximum likelihood framework, this procdedure is similar to bootstrapping. In fact,
314 the ultimate effect of this is to decrease the bootstrap confidence to a level that is more
315 in line with the measured uncertainty in the base calls.

316 In a Bayesian framework, the multinomial sampling could be incorporated as prior dis-
317 tribution on each nucleotide. For large collections of large sequences in an Markov Chain
318 Monte Carlo algorithm, this increases the dimensionality dramatically.

3   Applications
319

3.1   SARS-CoV-2 lineage assignemnt
320

321 In this section, we apply the re-sampling method to evaluate the impact of sequencing
322 error on the lineage assignments of SARS-CoV-2. Sequences are sampled from $\mathcal{S}'$ and then
323 assigned a lineage based on the current state-of-the-art in phylogenetic analyis.

324 We use the lineage designation algorithm described in Rambaut et al. (2020) and as-
325 sign our sequences to lineages using the pangoLEARN tool (Pangolin version 2.3.2,
326 pangoLEARN version 2021-02-21) that the authors have made available (github.com/

sampled_bars

Figure 1: Distribution of called lineages from Pangolin. Red bars indicate the lineage of the most probable sequence. Any lineage with fewer than 100 observations was grouped into the "Other" category.

cov-lineages/Pangolin). This tool uses a decision tree model to determine which lineage a given sequence is most likely to belong to. We demonstrate that even the best available tools are underestimating the variance and therefore producing overconfident conclusions.

3.1.1 Data

The data for this application were downloaded from NCBI's SRA web interface (https://www.ncbi.nlm.nih.gov/sra/?term=txid2697049). Search results were filtered to only include records that had SAM files. To select which runs to download, an arbitrary selection of 5-10 records from each of 20 non-sequential results pages were chosen. Once collecting the run accession numbers from the search results, an R script was run to download the relevant files and check that all information was complete. 23 out of 275 files were labelled incomplete due to having too few reads (possibly because the download timed out) or not containing any CIGAR strings. The GISAID accession numbers for the seqeunces we used are provided in the appendix. TODO: Other info about the files? TODO: add the accessions to the appendix

3.1.2 Re-sampling the probabilistic sequence

Since pangoLEARN is a pre-trained model, assigning lineage designations to a large number of resampled genome sequences is not computationally burdensome. Sampling 5,000 different sequences from a probabilistic sequence can be done in a reasonable amount of time, even on a mid-range consumer laptop.

For this analysis we use the most basic resampling strategy described above. We sample base calls from the multinomial distribution then use pangoLEARN to determine the lineage assignment.

TODO: Maybe move this to the figure legend? Or parts of it? In 1, each row represents 10,000 resamples from one SAM file. TODO: It's 10,000 resamples, right? Each bar represents one lineage assignment from Pangolin. The width of the bars represents the number times a re-sampled sequence resulted in the labelled lineage designation. There were 95 sequences total, but we only plotted ones where the second most common lineage designation had more than 250 observations. TODO: exact number of plots (I think it's 52 or something). Results for all lineages can be found in Table **??** in the Appendix.

Figure 1 shows that the consensus sequence almost always is assigned the same lineage as the majority of the resamples; the full results are in Table **??**. The proportion of resamples with the same lineage as the consensus sequence is very rarely 100% and can be as low as 22.09%. TODO: Provide accession number. It is noteworthy that the only times where 100% of resampled sequences agreed are when the lineage call was "None" or for the lineage labelled B.1.1.7, which is a significantly more infectious lineage and is of special concern to health authorities. TODO: make sure this is still true!; newer versions of Pangolin use post-processing of B.1.1.7 according to Art.

Initial analyses used Pangolin version 2.0.8, which used multinomial logistic regression, rather than decision trees, to assign lineages. This procedure resulted in bootstrap support values that varied from 0.6 to 1. After upgrading to Pangolin version 2.3.2, which uses a decision tree model, all bootstrap probabilities are reported as 1. However, the research team behind Pangolin maintain a record of diagnostics for their classification system (data/lineage_recall_report.csv in https://github.com/cov-lineages/pangoLEARN).

³⁷¹ This report includes the precision, recall, and F-value for each lineage in their classifica-
³⁷² tion system. We have included the F values of the for the lineage that the conseq was
³⁷³ assigned to in Table **??**.

3.2   Clock rate estimation for SARS-CoV-2

³⁷⁵ The clock rate (slope of a root-to-tip regression) for SARS-CoV-2 is commonly estimated
³⁷⁶ as a fixed rate near 0.001 mutations per site per year (specific estimates given below).
³⁷⁷ Using the same resampling methods as above, we estimate a clock rate for trees estimated
³⁷⁸ from each of 50 resamples and for the tree estimated based on the reported conseqs.

³⁷⁹ To obtain the data, we sampled genomes uniformly from each month of recorded data in
³⁸⁰ GenBank, using filters to ensure that the genomes were complete and had an associated
³⁸¹ SAM file. We further had to filter out SAM files that were incomplete or did not contain
³⁸² the CIGAR strings necessary for alignment. The associated GISAID accession numbers
³⁸³ are provided in the Appendix. TODO: add accession numbers.

³⁸⁴ Our re-sampling method will, by definition, introduce other possible mutations beyond
³⁸⁵ what the conseq suggests. Because of this, the apparent number of mutations between a
³⁸⁶ re-sampled genome and the estimated root is a function of the coverage, with more posi-
³⁸⁷ tions read or more uncertainty in the sequence leading to artificially inflated phylogenetic
³⁸⁸ branch lengths. This implies that the estimates of the time for the most recent common
³⁸⁹ ancestor are not reliable. However, assuming that the sequences have comparable levels
³⁹⁰ of uncertainty, each branch increases by a similar amount and the clock rate should not
³⁹¹ be affected.

³⁹² The sequences that we acquired did not have comparable levels of uncertainty; the viruses
³⁹³ sampled early in the pandemic had considerable higher uncertainty. To account for this,
³⁹⁴ we found the sum of $\mathcal{S}'$ for each sequence and applied Statistical Process Control tech-
³⁹⁵ niques to ensure that all of the sequences had a similar level of coverage. In particular,
³⁹⁶ we found the mean of the coverage of the sequences in our data set, $\bar{c}$, and the standard
³⁹⁷ deviation of the coverage, s. We removed any sequences outside of $\bar{x} \pm 3s$, recalculated
³⁹⁸ $\bar{x}$ and s, and iterated the removal process until all sequence coverages were within the
³⁹⁹ bounds.

⁴⁰⁰ The clock rate was estimated using TreeTime (Sagulenko et al., 2018). We recorded
⁴⁰¹ the clock rate and standard error from the time tree constructed using the consensus
⁴⁰² sequences and compared this to the clock rate and standard deviations of the estimated
⁴⁰³ clock rates in the resampled sequences. The tree built from conseqs had a clock rate of
⁴⁰⁴ 6.5×10^{-4} with a standard error of 8.01×10^{-5}. The mean of the clock rates for all of the
⁴⁰⁵ sets of resampled sequences was 8.6×10^{-4} with standard deviation of 5.3×10^{-4}, which
⁴⁰⁶ is approximately 1.6 times as large as the standard error for the conseqs.

⁴⁰⁷ The estimates of the clock rate are shown in Figure 2. The red line and shaded region
⁴⁰⁸ are the clock rate for the tree built from conseqs along with ± 1.96 standard errors. Rate
⁴⁰⁹ estimates from Duchene et al. (2020) (n=122), Choudhary et al. (2021) (n=261), Song
⁴¹⁰ et al. (2021) (n=29), Nie et al. (2020) (n=112), and Geidelberg et al. (2021) (n=77) are
⁴¹¹ also labelled on the plot with purple error bars for 95% Bayesian Credible Intervals (BCI)
⁴¹² or 95% Highest Posterior Density (HPD), indicating that the rates and errors from each
⁴¹³ root-to-tip regression are in line with other published results. Figure 2 demonstrates that
⁴¹⁴ the estimated evolutionary rates have an average close to the rate estimated from the
⁴¹⁵ conseqs, but each of the individual error bars miss the excess variation due to sequence
⁴¹⁶ uncertainty.

RTT_Slope

Figure 2: Clock rates (slope) and 95% Confidence Intervals for the collections of re-sampled sequences. The red line and red shaded region are the clock rate and 95% CI for the consensus sequences. The purple points and error bars are the clock rates and error intervals (either Bayesian Credible Interval or Highest Posterior Probability) from published studies, as labelled. The re-sampled sequences are in line with the consensus sequences as well as the published sequences, but represent a much larger variation due to the uncertainty in the original genome sequences.

4   Conclusions

The short run files produced by next generation sequencing platforms include valuable information about the quality of base calls which can easily be propagated into analyses. In this study, we demonstrated that these errors in base calling can lead to different conclusions when determining a lineage via Pangolin and that the variance in clock rate estimates is larger than previously shown due to these errors. Both of these situations could lead to incorrect conclusions, such as missing a variant of interest or making overconfident conclusions about the date of the first case of COVID-19. The potential for errors in base calls should always be taken into account when making decisions based on genetic sequencing data.

The primary contribution of this research is the construction of the probability sequence, which allows for a wide variety of future research directions. The direction we described here is focused on re-sampling, which allows a more complete appraisal of the variance in the estimates (or provides a reasonable prior distribution in a Bayesian setting), while comparing results for the most likely sequences provide a measure of robustness to sequence uncertainty. However, this is not the only application of the probabilistic sequence.

Instead of SAM files, FASTQ files are often the best available quantification of the potential sequence read errors. We have already shown that probability sequences can be trivially constructed by spreading the remaining error probability uniformly across the remaining bases. Another use of FASTQ files would be to construct a probability sequence as a reference genome for a given category. This would entail collecting all available FASTQ files for a given lineage designation and using them in the construction of a probability sequence as if they were short reads in a SAM file. From here, lineage designation for a newly acquired sequence (and it's probability sequence) could be performed via a hypothesis test for whether the probability sequences are sufficiently similar.

Our proposed methods can result in a linear increase in computational expense. Even the method based on ordering the sequences by likelihood inevitably requries re-running the analysis numerous times. TODO: delete? However, we have demonstrated that the uncertainty in the sequences themseleves can lead to major changes to the interpretations of the results. The so-called "consensus sequence" is simply the most likely sequence, and the reported uncertainty is not merely an academic curiousity. Ideally individual analyses would be constructed to take nucleotide-level uncertainty into account. For instance, phylogenies have been estimated based on uncertain sequence information in Ross and Markowetz (2016); Jahn et al. (2016); Zafar et al. (2017) but the uncertainty is not derived from base quality scores. An extension of these methods to incorporate the base quality scores may be a worthwhile research direction.

Computational burden can also be reduced by ording the sequences according to the sequence level uncertainty discussed above. By putting all possible sequences in order of uncertainty, we can analyse how the biggest changes in uncertainty affect our analysis. It is possible to devise an algorithm that puts the sequences in order of their uncertainty without calculating the uncertainty for every sequence. The conseq is clearly the least uncertain sequence, and by the construction of the probability sequence, it is computa-

460 tionally easy to calculate sequence uncertainty relative to the conseq. Because of this, it
461 should be reasonable to calculate all sequence uncertainties for a single mutation, then all
462 pairs of mutations, and so on (we note that it is possible for some pairs of mutations to be
463 more likely than single mutations, thus this algorithm cannot be perfect). This allows for
464 easy sorting of the most likely sequences, assuming enough sequence uncertainties have
465 been calculated. With this sorted list, we can re-fit a model at various levels of uncertainty
466 as a way to test the robustness of a given conclusion to sequence uncertainty.

467 Our analysis focused on designation of lineages according to the Pangolin model as well
468 as estimation of the clock rate. The importance of incorporating sequence uncertainty
469 is not confined to these applications; any analysis involving sequenced genomes would
470 benefit from some method of incoporating the uncertainty or including some measure of
471 robustness. For example, the estimated frequency of alleles in the population could be
472 used as the probability sequence, then propagated into further analysis.

473 Our method does not preclude tertiary analyses to test for systematic errors or deviations
474 from a Mendelian inheritance pattern assumption. We cannot account for systematic
475 errors, such as those present due to human errors (*e.g.,* human error, as noted in Iss,
476 2020). Our method allows for adjustments of the base call quality score, such as in
477 Brockman et al. (2008), as well as more sophisticated definitions of genome likelihoods
478 (*e.g.,* Li et al., 2004; DePristo et al., 2011; Li et al., 2009b).

479 We have developed an algorithm to include insertion events in a re-sampling scheme, but
480 many of the resultant sequences were not mappable to known sequences. The Pangolin
481 lineage assignment system appears to treat insertions differently from single nucleotide
482 polymorphisms, and our method of sampling insertions is incompatible with their treat-
483 ment of insertions in lineage assignment operations. This is potentially because the sam-
484 pled base pair at any given position is independent of each other position, and the inser-
485 tions observed in real-world data are possibly always associated with particular mutations
486 elsewhere.

487 This study should not be taken in any way as a criticism of the Pangolin lineage assignment
488 procedure. Rather, Pangolin was chosen as it is a state-of-the art tool based on the best
489 available algorithms for phylogeny reconstruction. The phylogeny created by this team
490 has been a vital resource for researchers and for public health professionals. In particular,
491 their label for the current Variants of Concern (VOCs), especially B.1.1.7, are the labels
492 being used worldwide by news organizations.

493 TODO: ordered likelihood conclusions / usefulness as a way of approximating the distri-
494 bution (kinda but not really like a Taylor series).

References

496 (2020). Issues with SARS-CoV-2 sequencing data - SARS-CoV-2 coronavirus / nCoV-2019
497   Genomic Epidemiology. https://virological.org/t/issues-with-sars-cov-2-sequencing-
498   data/473.

499 Beerenwinkel, N. and Zagordi, O. (2011). Ultra-deep sequencing for the analysis of viral
500   populations. *Current Opinion in Virology*, 1(5):413–418.

501 Brockman, W., Alvarez, P., Young, S., Garber, M., Giannoukos, G., Lee, W. L., Russ, C.,
502   Lander, E. S., Nusbaum, C., and Jaffe, D. B. (2008). Quality scores and SNP detection
503   in sequencing-by-synthesis systems. *Genome Research*, 18(5):763–770.

504 Choudhary, M. C., Crain, C. R., Qiu, X., Hanage, W., and Li, J. Z. (2021). Severe
505   Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) Sequence Characteristics of
506   Coronavirus Disease 2019 (COVID-19) Persistence and Reinfection. *Clinical Infectious
507   Diseases*, (ciab380).

Clement, N. L., Snell, Q., Clement, M. J., Hollenhorst, P. C., Purwar, J., Graves, B. J., Cairns, B. R., and Johnson, W. E. (2010). The GNUMAP algorithm: Unbiased probabilistic mapping of oligonucleotides from next-generation sequencing. *Bioinformatics*, 26(1):38–45.

DePristo, M. A., Banks, E., Poplin, R., Garimella, K. V., Maguire, J. R., Hartl, C., Philippakis, A. A., del Angel, G., Rivas, M. A., Hanna, M., McKenna, A., Fennell, T. J., Kernytsky, A. M., Sivachenko, A. Y., Cibulskis, K., Gabriel, S. B., Altshuler, D., and Daly, M. J. (2011). A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–498.

Doronina, N. V. (2005). Phylogenetic position and emended description of the genus Methylovorus. *INTERNATIONAL JOURNAL OF SYSTEMATIC AND EVOLUTIONARY MICROBIOLOGY*, 55(2):903–906.

Duchene, S., Featherstone, L., Haritopoulou-Sinanidou, M., Rambaut, A., Lemey, P., and Baele, G. (2020). Temporal signal and the phylodynamic threshold of SARS-CoV-2. *Virus Evolution*, 6(2).

Ewing, B. and Green, P. (1998). Base-Calling of Automated Sequencer Traces Using *Phred*. II. Error Probabilities. *Genome Research*, 8(3):186–194.

Fuller, C. W., Middendorf, L. R., Benner, S. A., Church, G. M., Harris, T., Huang, X., Jovanovich, S. B., Nelson, J. R., Schloss, J. A., Schwartz, D. C., and Vezenov, D. V. (2009). The challenges of sequencing by synthesis. *Nature Biotechnology*, 27(11):1013–1023.

Fumagalli, M., Vieira, F. G., Korneliussen, T. S., Linderoth, T., Huerta-Sánchez, E., Albrechtsen, A., and Nielsen, R. (2013). Quantifying Population Genetic Differentiation from Next-Generation Sequencing Data. *Genetics*, 195(3):979–992.

Geidelberg, L., Boyd, O., Jorgensen, D., Siveroni, I., Nascimento, F. F., Johnson, R., Ragonnet-Cronin, M., Fu, H., Wang, H., Xi, X., Chen, W., Liu, D., Chen, Y., Tian, M., Tan, W., Zai, J., Sun, W., Li, J., Li, J., Volz, E. M., Li, X., and Nie, Q. (2021). Genomic epidemiology of a densely sampled COVID-19 outbreak in China. *Virus Evolution*, 7(1).

Gompert, Z. and Buerkle, C. A. (2011). A Hierarchical Bayesian Model for Next-Generation Population Genomics. *Genetics*, 187(3):903–917.

Jahn, K., Kuipers, J., and Beerenwinkel, N. (2016). Tree inference for single-cell data. *Genome Biology*, 17(1):86.

Keith, J. M., Adams, P., Bryant, D., Kroese, D. P., Mitchelson, K. R., Coachran, D. A. E., and Lala, G. H. (2002). A simulated annealing algorithm for finding consensus sequences. *Bioinformatics*, 18(11):1494–1499.

Kozlov, O. (2018). Models, Optimizations, and Tools for Large-Scale Phylogenetic Inference, Handling Sequence Uncertainty, and Taxonomic Validation.

Kuhner, M. K. and McGill, J. (2014). Correcting for Sequencing Error in Maximum Likelihood Phylogeny Inference. *G3 Genes—Genomes—Genetics*, 4(12):2545–2552.

Kuo, T., Frith, M. C., Sese, J., and Horton, P. (2018). EAGLE: Explicit Alternative Genome Likelihood Evaluator. *BMC Medical Genomics*, 11(2):28.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., and Durbin, R. (2009a). The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16):2078–2079.

Li, H., Ruan, J., and Durbin, R. (2008). Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Research*, 18(11):1851–1858.

Li, M., Nordborg, M., and Li, L. M. (2004). Adjust quality scores from alignment and improve sequencing accuracy. *Nucleic Acids Research*, 32(17):5183–5191.

Li, R., Li, Y., Fang, X., Yang, H., Wang, J., Kristiansen, K., and Wang, J. (2009b). SNP detection for massively parallel whole-genome resequencing. *Genome Research*, 19(6):1124–1132.

Lippi, G., Simundic, A.-M., and Plebani, M. (2020). Potential preanalytical and analytical vulnerabilities in the laboratory diagnosis of coronavirus disease 2019 (COVID-19). *Clinical Chemistry and Laboratory Medicine*, 58(7):1070–1076.

NC-IUB (1986). Nomenclature for incompletely specified bases in nucleic acid sequences. Recommendations 1984. Nomenclature Committee of the International Union of Biochemistry (NC-IUB). *Proceedings of the National Academy of Sciences of the United States of America*, 83(1):4–8.

Nie, Q., Li, X., Chen, W., Liu, D., Chen, Y., Li, H., Li, D., Tian, M., Tan, W., and Zai, J. (2020). Phylogenetic and phylodynamic analyses of SARS-CoV-2. *Virus Research*, 287:198098.

O'Rawe, J. A., Ferson, S., and Lyon, G. J. (2015). Accounting for uncertainty in DNA sequencing data. *Trends in Genetics*, 31(2):61–66.

Rambaut, A., Holmes, E. C., O'Toole, Á., Hill, V., McCrone, J. T., Ruis, C., du Plessis, L., and Pybus, O. G. (2020). A dynamic nomenclature proposal for SARS-CoV-2 lineages to assist genomic epidemiology. *Nature Microbiology*.

Richterich, P. (1998). Estimation of Errors in "Raw" DNA Sequences: A Validation Study. *Genome Research*, 8(3):251–259.

Robasky, K., Lewis, N. E., and Church, G. M. (2014). The role of replicates for error mitigation in next-generation sequencing. *Nature Reviews Genetics*, 15(1):56–62.

Ross, E. M. and Markowetz, F. (2016). OncoNEM: Inferring tumor evolution from single-cell sequencing data. *Genome Biology*, 17(1):69.

Sagulenko, P., Puller, V., and Neher, R. A. (2018). TreeTime: Maximum-likelihood phylodynamic analysis. *Virus Evolution*, 4(1):vex042.

Schneider, T. D. (2002). Consensus Sequence Zen. *Applied bioinformatics*, 1(3):111–119.

Schneider, T. D. and Stephens, R. (1990). Sequence logos: A new way to display consensus sequences. *Nucleic Acids Research*, 18(20):6097–6100.

Song, N., Cui, G.-L., and Zeng, Q.-L. (2021). Genomic Epidemiology of SARS-CoV-2 From Mainland China With Newly Obtained Genomes From Henan Province. *Frontiers in Microbiology*, 12:673855.

Stormo, G. D., Schneider, T. D., Gold, L., and Ehrenfeucht, A. (1982). Use of the 'perceptron' algorithm to distinguish translational initiation sites in E. coli. *Nucleic Acids Research*, 10(9):16.

Zafar, H., Tzen, A., Navin, N., Chen, K., and Nakhleh, L. (2017). SiFit: Inferring tumor trees from single-cell sequencing data under finite-sites models. *Genome Biology*, 18(1):178.