Sequencing Uncertainty Propagation

David Champredon

November 11, 2019

1 Background

Identifying the sequence of nucleotides from a biological sample is a complex process which is fraught with noise.

Assuming the biological sample of interest has been properly isolated (that is it is complete, has no damages or contamination), sequencing a biological sample, whether with the Sanger method or "Next-Generation Sequencing" (NGS) usually involves:

- if the sample of interest is RNA: Reverse transcription
- DNA fragmentation in smaller pieces than the original sample (less than 400bp for NGS, 800bp for Sanger)
- amplification of the fragmented DNA using PCR
- sequencing the fragments (identifying the nucleotides from a fluorescent tag attached)
- alignment or mapping: putting back the small fragment together by aligning them (de novo) or mapping them on benchmark libraries

Errors can be introduced at each of these steps for various reasons [1]. It is probably not feasible to determine what is the source of the noise, nor to try to eliminate it completely. The goal here is to acknowledge there is uncertainty in the output sequence given from any sequencing method and to propose a method to propagate this uncertainty in any downstream analysis. Currently, this uncertainty is recognized and even quantified with sequencing quality scores (FASTQ files), but it does not

seem those scores are used to inform a probabilistic model to represent the sequence. Simply put, we shouldn't treat the result of sequencing as a *certainty*.

2 Probabilistic representation

2.1 Definition

We can represent probabilistically a nucleotide sequence in a matrix form. For a sequence of length ℓ we can write:

$$S = \begin{array}{c} \\ \texttt{A} \\ \texttt{C} \\ \texttt{G} \\ \texttt{T} \\ \texttt{x} \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & \dots & \ell \end{array} \\ \begin{pmatrix} p_{A,1} & p_{A,2} & \dots & p_{A,\ell} \\ p_{C,1} & p_{C,2} & \dots & p_{C,\ell} \\ p_{G,1} & p_{G,2} & \dots & p_{G,\ell} \\ p_{T,1} & p_{T,2} & \dots & p_{T,\ell} \\ p_{x,1} & p_{x,2} & \dots & p_{x,\ell} \end{pmatrix} \end{array}$$

Each column represents the nucleotide position, each row one of the four nucleotide `A,C,G,T` as well as an empty position `x`. Hence, S is a $5 \times \ell$ matrix. Its elements represent the probability that a nucleotide is at given position:

$$S_{n,j} = \Pr(\text{nucleotide } \texttt{n} \text{ is at position } j) \tag{1}$$

with the special case for a deletion:

$$S_{\texttt{x},j} = \Pr(\text{empty position } j) \tag{2}$$

The matrix S will be called the *probabilistic sequence* of a biological sample. Note, that we have, for all $1 \leq j \leq \ell$:

$$\sum_{n \in \{\texttt{A,C,G,T,x}\}} S_{n,j} = 1 \tag{3}$$

The sequence of nucleotides from a biological sample is not treated as certainty anymore, but as a collection of possible sequences, with length not necessarily equal when the probability of an empty position is positive.

2.2 Examples

If we have the following probabilistic sequence

$$S = \begin{pmatrix} 0.9 & 0.05 & 0.99 & 0 & 0 \\ 0 & 0.8 & 0 & 0 & 0.1 \\ 0.1 & 0.15 & 0 & 0.3 & 0.9 \\ 0 & 0 & 0.01 & 0.7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

then there are $2 \times 3 \times 2^3 = 48$ possible sequences. The most likely is the one having the highest nucleotides probabilities: `ACATG` with probability 0.449 ($0.9 \times 0.8 \times 0.99 \times 0.7 \times 0.9$).

If there is a positive probability for at least one empty position, then the sequence has a variable length. Let's take the same example as above, but adding one possible empty position:

$$S = \begin{pmatrix} 0.9 & 0.05 & 0.99 & 0 & 0 \\ 0 & 0.8 & 0 & 0 & 0.1 \\ 0.1 & 0.15 & 0 & 0.2 & 0.9 \\ 0 & 0 & 0.01 & 0.7 & 0 \\ 0 & 0 & 0 & 0.1 & 0 \end{pmatrix}$$

Like above, there is still a 0.449 probability that the sequence is `ACATG`, but with probability 0.064, the sequence could be shorter when position 4 is empty and be `ACAG`.

3 Quantifying probabilities

The difficulty of estimating the probabilities that populate the probabilistic sequence lies in quantifying errors at each steps (fragmentation, amplification, sequencing, alignment). And how to integrate these error types in a coherent fashion?

3.1 Sequencing errors

Maybe the easiest error type to quantify is the fragment sequencing error because a quality (or "Phred") score is attributed to each base call. The quality score Q is directly related to the error probability: $p = 10^{-Q/10}$ (Figure 1).

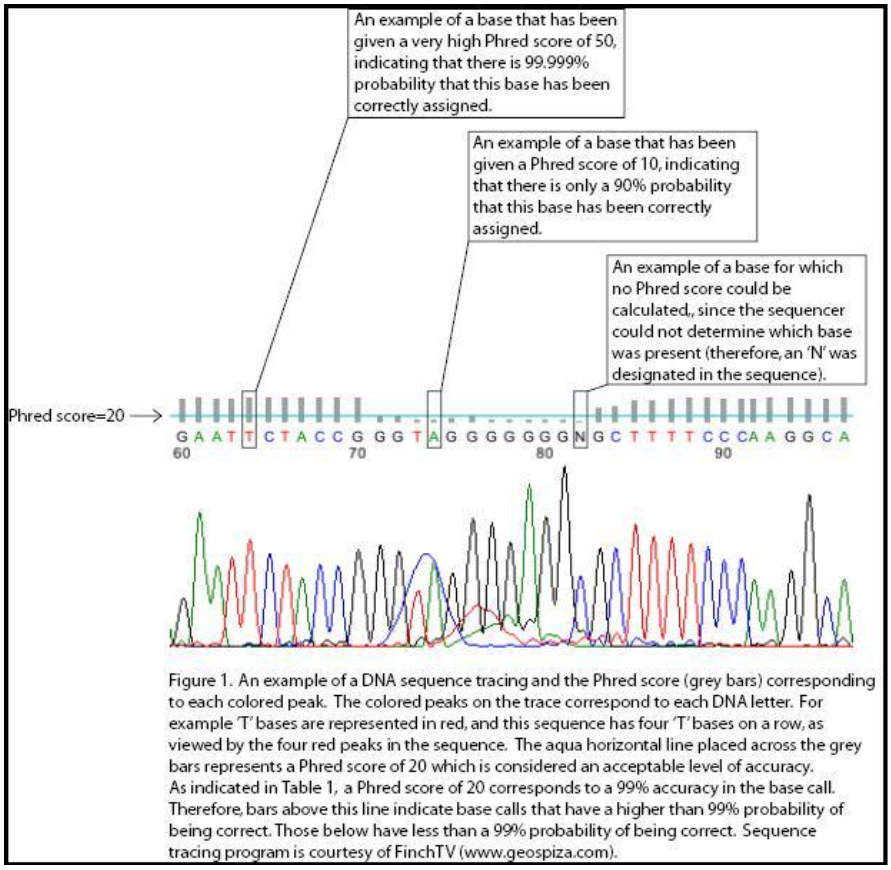

Figure 1: Example of quality scores associated with a chromatograph.

As I don't have FASTQ files from actual biological samples, I use the software inSilicoSeq that simulates the sequencing of short reads from Illumina instruments. As shown in Figure 2, the sequencing error probability ranges between $10^{-3.5}$ and $10^{-1.5}$.

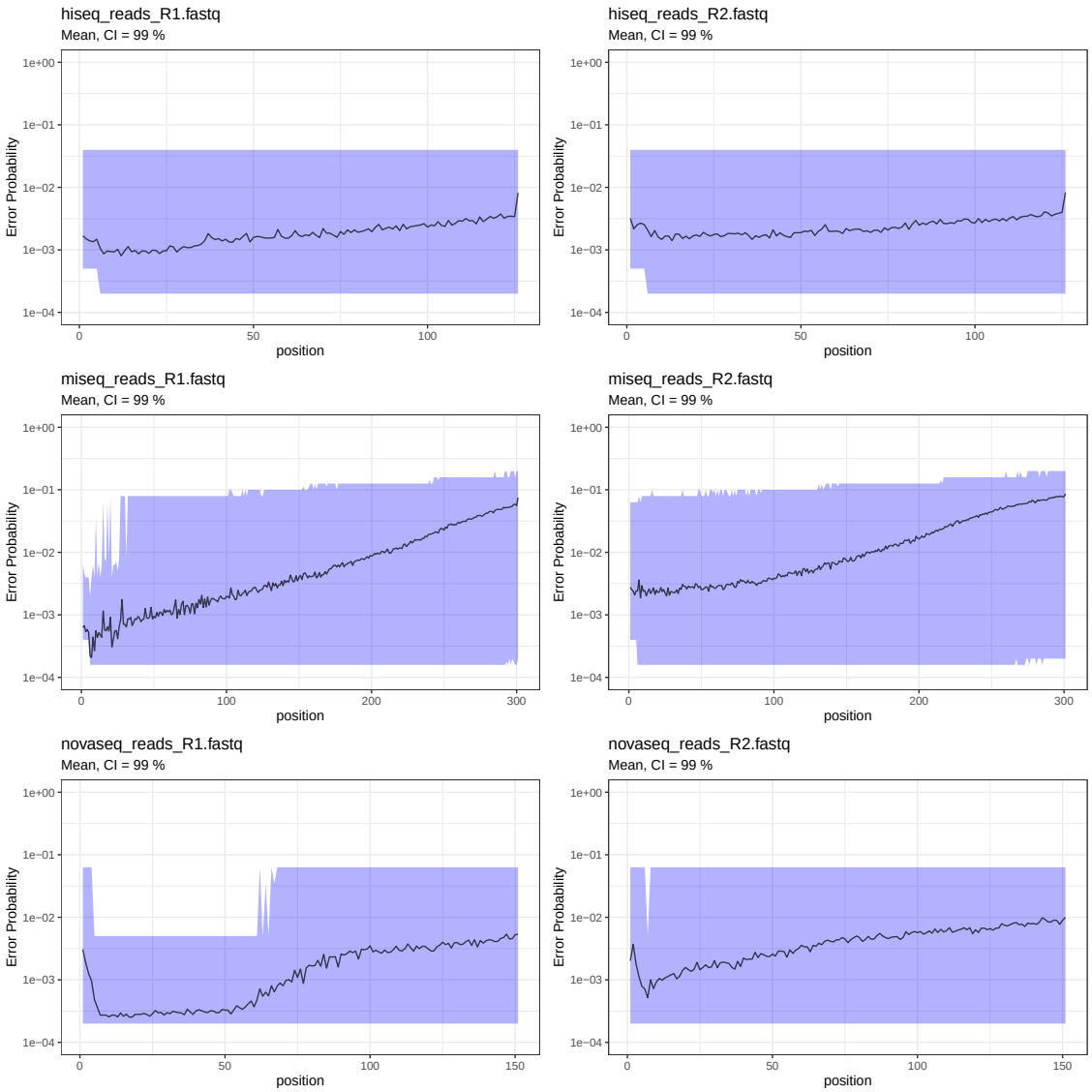

Figure 2: Error probability of calling bases for 3 different Illumina instruments (HiSeq, MiSeq, NovaSeq) simulated with InSilicoSeeq. Figure generated by running `reads-seq-err/go.sh`.

After alignment, taking an optimistic view, we can assume a unique global sequencing error of $\epsilon = 10^{-3.5} = 0.0003$. So each base call is right with probability $1 - \epsilon$. Assume the other bases and missing position x are all equally likely with probability

$(1-\epsilon)/4$.

For example, if the output sequence after fragment sequencing and alignment is `ACGT` the probabilistic sequence is:

$$S = \begin{pmatrix} 1-\epsilon & \epsilon/4 & \epsilon/4 & \epsilon/4 \\ \epsilon/4 & 1-\epsilon & \epsilon/4 & \epsilon/4 \\ \epsilon/4 & \epsilon/4 & 1-\epsilon & \epsilon/4 \\ \epsilon/4 & \epsilon/4 & \epsilon/4 & 1-\epsilon \\ \epsilon/4 & \epsilon/4 & \epsilon/4 & \epsilon/4 \end{pmatrix}$$

3.2 Polymorphism, in-host diversity

The probabilistic sequence can also be interpreted as a representation of polymorphism abundance in a biological sample from one single host (in-host diversity).

There may be one problem: the data will give abundances of several sequences – say 10 – found in one host. When sampling from the probabilistic sequence, it is not constraint to a given number of specific sequences. For a sequence of length n the number of unique sequences is 5^n. Even if we do not observe perfectly, it is almost sure that the real diversity is a minute fraction of the 5^n possible sequences.

Maybe this can be overcome by thinking in terms of diversity *and* sequencing error. In that case the probabilities would be defined primarily for the diversity and another smaller layer for sequencing error.

Not too sure about all that...

4 Sequence logo and entropy

Are sequence logos along with their entropy calculation a better start than the probabilistic sequence?

Schneider [2, 4, 3] used information theory, Shannon entropy, to quantify the uncertainty in "bits". Is that better?

The bits are the number of binary digits, i.e., 0 or 1, needed to translate the possibilities of a "message". Here the message is a base call, or rather bases associated with their probabilities. The larger the number of bits required to translate the message, the more uncertain. In my mind, uncertainty and entropy are similar, the only important difference is that entropy is quantified in bits.

The entropy (i.e., uncertainty) at a given position j is given by Shannon's standard expression:

$$H = -\sum_{n \in \{\texttt{A},\texttt{C},\texttt{G},\texttt{T}\}} S_{n,j} \log_2(S_{n,j}) \tag{4}$$

with, as before, $S_{n,j}$ being the probability that nucleotide n is at position j.

For example, if we know for sure that the nucleotide at a given position is C, that is $S_{\texttt{C},j} = 1$, then entropy (uncertainty) is $H = 0$: we don't need any bit to translate that C is at this position, it's sure, we know that.

Another example, is when any of the four bases are equally likely. All probabilities are $1/4$, and the entropy is maximum with a value $H = 2$ bits $(= -4 \times \frac{1}{4} \times \log_2(\frac{1}{4}))$.

I'm not sure how useful Schneider's $R_{sequence}$ metric is in my case (propagating uncertainty in tree reconstruction). The entropy H may be more useful as a principled measure of uncertainty at a given position.

5 Generating simulated probabilistic sequences

If we want to simulate realistic probabilistic sequence, we have to reproduce a similar uncertainty as the one we would have from either sequencing error or polymorphism.

The data from Zanini [5] is an excellent source to assess primarily the diversity of polymorphism for HIV, and to a certain extent too, the sequencing error (because it is always here). This data gives, for several patients at several time points during their (untreated) infection the number of times nucleotides were sample at a given position, across the whole HIV genome. The number of nucleotide occurrences at each position can easily be transformed into the probabilities for the probabilistic sequence. The entropy can then be calculated at each position, and also for the entire genome (by simply summing up the entropies for all positions).

The entropy is a measure of uncertainty. So we can consider the distribution of entropies (one for each position on the genome) as a representation of the overall genome sequencing uncertainty, that should be approximately matched by simulations deemed realistic.

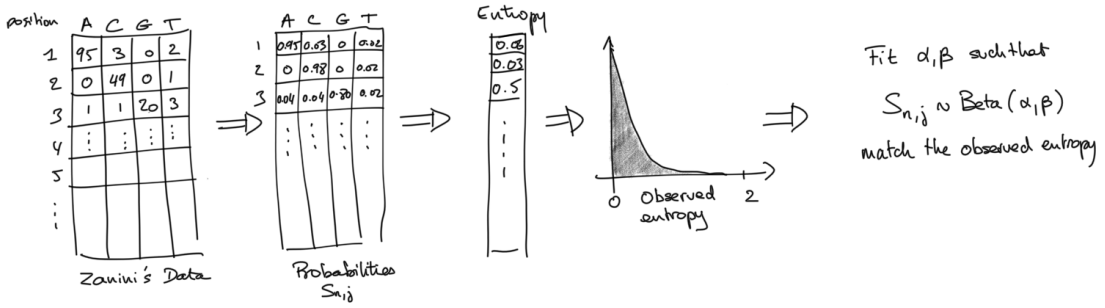

Figure 3: How to reproduce observed sequencing uncertainty.

The data from Zanini shows that $S_{n,j}$, the probability of nucleotide n to be at position j is highly concentrated close to (and below!) 1. So, if we want to simulate the those probabilities, we can draw them from a Beta distribution where the shape parameters α and β were fitted on the observed entropy distribution:

$$S_{n,j} \sim \text{Beta}\,(\alpha, \beta) \tag{5}$$

$$\alpha, \beta \text{ such that } E(\alpha, \beta) = E_{obs} \tag{6}$$

where E is the distribution of position-wise entropy.

A fit on Zanini's data gives approximately $\hat{\alpha} = 29.7$ and $\hat{\beta} = 0.06$. Hence, using $\hat{\alpha}$ and $\hat{\beta}$ in simulations of probabilistic sequences should give an "uncertainty profile" that is similar to the one observed in Zanini's data.

References

[1] Niko Beerenwinkel and Osvaldo Zagordi. Ultra-deep sequencing for the analysis of viral populations. *Current Opinion in Virology*, 1(5):413–418, 2011.

[2] T D Schneider, G D Stormo, L Gold, and A Ehrenfeucht. Information content of binding sites on nucleotide sequences. *J Mol Biol*, 188(3):415–431, Apr 1986.

[3] Thomas D Schneider. Consensus sequence zen. *Applied bioinformatics*, 1(3):111–119, 2002.

[4] Thomas D Schneider and R Michael Stephens. Sequence logos: a new way to display consensus sequences. *Nucleic acids research*, 18(20):6097–6100, 1990.

[5] Fabio Zanini, Johanna Brodin, Lina Thebo, Christa Lanz, Göran Bratt, Jan Albert, and Richard A Neher. Population genomics of intrapatient hiv-1 evolution. *Elife*, 4:e11282, 2015.