Supplementary Information

*E*stimating somatic mutation rates by duplex sequencing in non-model organisms: *Daphnia magna* as a case study

Eli Sobel, Jeremy E. Coate, and S. Schaack

Table of Contents

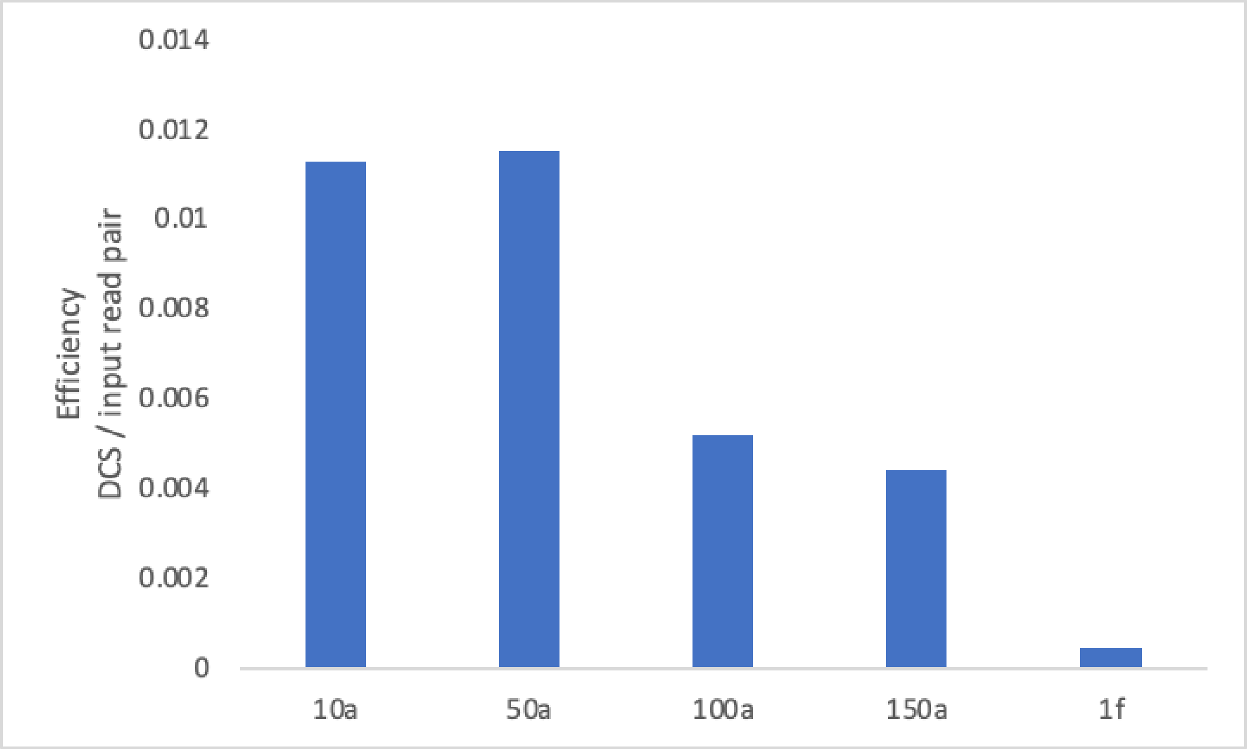

Figure S1 Consensus-making efficiency (DCS per input read pair) for five diluted libraries of *D. magna* gDNA without down-sampling. DCS were not filtered for insert size or alignment score.

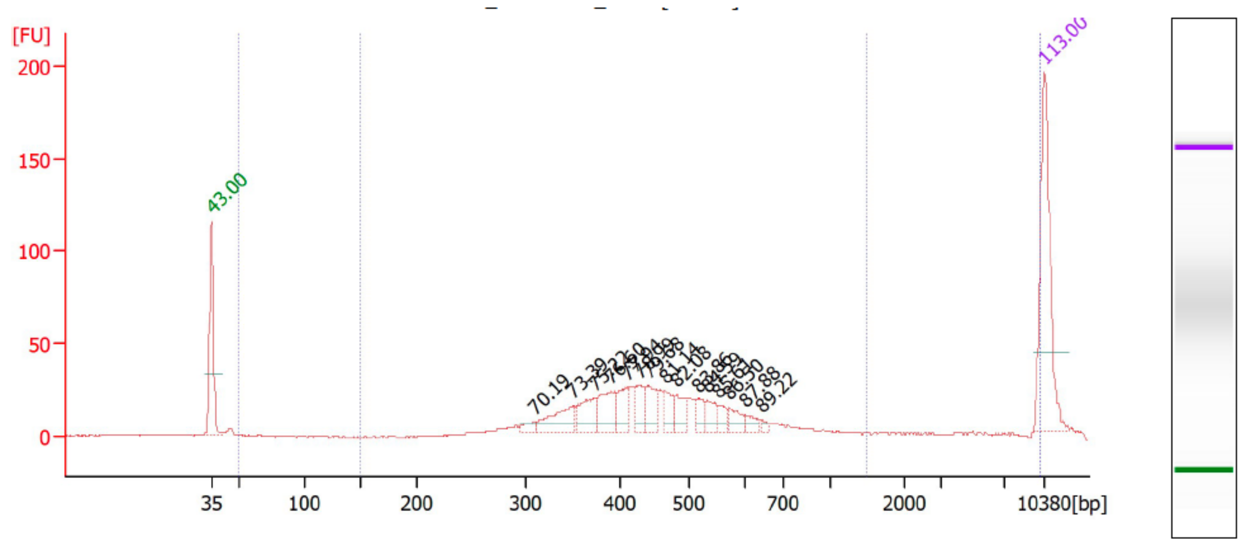

Figure S2 Bioanalyzer trace for the undiluted and unamplified NEBNext Ultra II FS library used as source for the 5 bottlenecked libraries (average fragment size estimated to be 456bp).

Estimates of genome copies per library

The undiluted and unamplified library that was used as input for the five bottlenecked libraries had an average fragment size of 456bp (**Fig. S2**). The number of molecules per diluted library was calculated as the molar mass of input DNA x Avogadro's number (6.022 x 10^{23}). The number of base pairs of DNA per library was then calculated by multiplying the number of molecules by the average length (456bp). Finally, the number of genome equivalents per library was estimated by dividing the total number of base pairs in the library by the number of base pairs in the *D. magna* genome. Two published estimates for the size of the *D. magna* genome vary considerably, so we estimated genome copy number using each - 123 Mbp [1] and 238 Mbp [2]. Values are shown in **Table S1**.

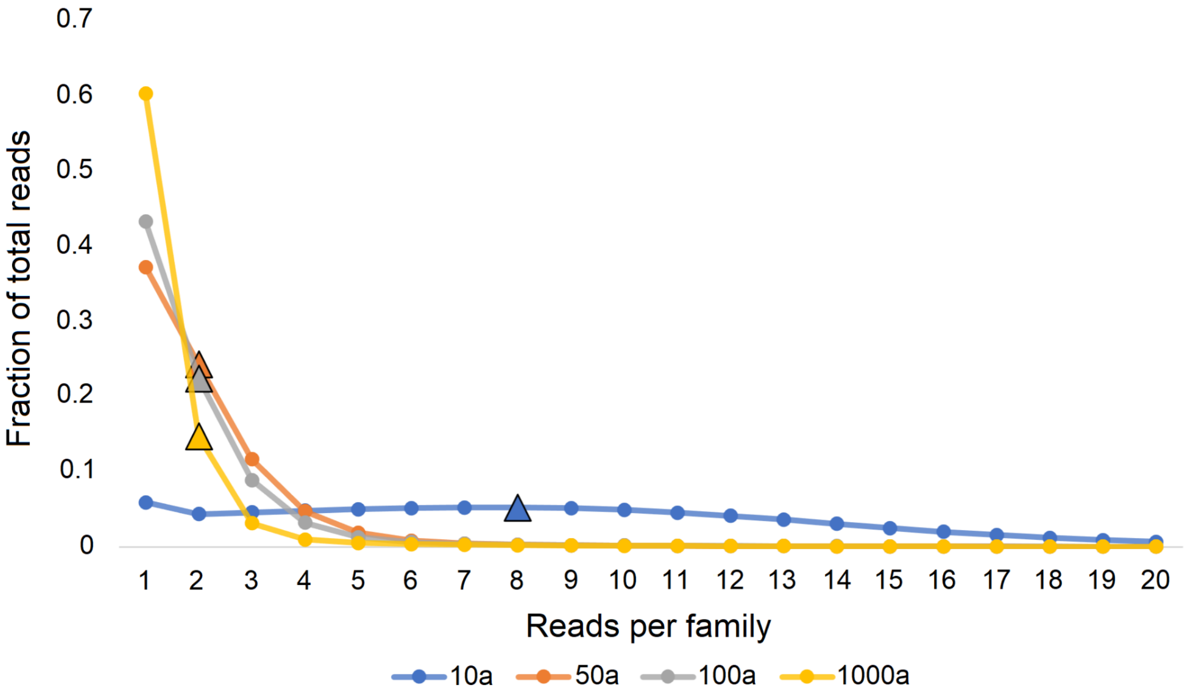

Figure S3 Distribution of read family sizes for four diluted libraries of *D. magna* gDNA, each downsampled to 50M reads. Colored triangles indicate the peak family size for each library.

Table S1 PCR primers and corresponding dual indices incorporated into the sequencing libraries. Primers are from the NEBNext Multiplex Oligos for Illumina (Dual Index Primer Set 1; NEB #7600S).

Library	i5 Primer	i5 Index Sequence	i7 Primer	i7 Index Sequence
10 amol	i504	TCAGAGCC	i708	TAATGCGC
50 amol	i505	CTTCGCCT	i709	CGGCTATG
100 amol	i508	GTCAGTAC	i712	AGCGATAG
150 amol	i506	TAAGATTA	i710	TCCGCGAA
1000 amol	i507	CTTCGCCT	i711	TCTCGCGC

Table S2 Parameters specified in the Configuration .csv for running the Duplex Sequencing pipeline.

sample	sample_name
rglb	name
rgpl	Illumina
rgpu	name
rgsm	name
reference	/path/to/reference.fasta
target_bed	/path/to/target.bed
blast_db	NONE
targetTaxonId	35525
baseDir	base_dir_name
in1	Raw_R1.fastq
in2	Raw_R2.fastq
mqFilt	30
minMem	2
maxMem	200
cutOff	1
nCutOff	0.02
umiLen	0
spacerLen	0
locLen	12
readLen	150
clipBegin	15
clipEnd	0
minClonal	0
maxClonal	1
minDepth	1
maxNs	1
runSSCS	false
recovery	noRecovery_noSynLink.sh
adapterSeq	/path/to/TruSeq3-PE-2.fasta

Table S3 Mutation frequencies for all 126,670 unfiltered variants, the 14 variants remaining after filtering, and the germline mutation frequencies generated by Ho et al. [10] for comparison.

Mutation type	Unfiltered	Final 14	Germline
C>A	0.127	0.500	0.126
C>G	0.064	0.071	0.067
C>T	0.280	0.143	0.283
T>A	0.155	0.143	0.15
T>C	0.258	0.143	0.261
T>G	0.115	0	0.113
Transitions	0.538	0.286	0.544
Transversions	0.462	0.714	0.456

Note: Due to the very small sample size (n=14), the frequencies of the final 14 variants remaining after filtering should not be considered to be statistically significant.

Table S4 Estimates of the number of genome equivalents per DNA library.

				# genome equivalents[3]	
Input DNA	Input DNA (moles)	# molecules[1]	# BPs[2]	Routtu et al. 2014	Lee et al. 2019
1 fmole	1E-15	6.02E+08	2.35E+11	986.8	1909.4
150 amoles	1.5E-16	9.03E+07	3.52E+10	148.0	286.4
100 amoles	1.00E-16	6.02E+07	2.35E+10	98.7	190.9
50 amoles	5.00E-17	3.01E+07	1.17E+10	49.3	95.5
10 amoles	1.00E-17	6.02E+06	2.35E+09	9.9	19.1

[1] # molecules = moles * 6.022E23

[2] # BPs = # molecules * 390bp/molecule (average fragment size = 456bp minus 66bp of adapater, leaving 390bp of genomic DNA)

[3] # genome equivalents = # BPs / genome size (bp) [Values are given for a genome size estimate from flow cytometry (238 Mbp; [2]), and from a recently published genome sequence (123 Mbp; [1])

Potential sources of error, and modifications to, Duplex Sequencing to further refine somatic mutation rate estimates

By randomly surveying the entire genome in a mostly unbiased fashion, and not restricting sequencing to small target regions, bottleneck sequencing has the advantage of not requiring exogenous UMIs. When randomly surveying a typical eukaryotic genome, it is extremely unlikely that two sampled DNA fragments will have sheared at the exact same genomic coordinates. Consequently, one can use the end sequences and mapping coordinates of reads as molecular identifiers to group read families (endogenous barcoding), rather than ligating UMI adapters to DNA fragments prior to sequencing (exogenous barcoding). By enabling the use of endogenous UMIs, bottlenecked Duplex Sequencing streamlines library preparation. However, the likelihood that two copies of a given DNA molecule will have the same breakpoints ("overlap by accident" or "OBA"; [3]) increases with decreasing amounts of DNA examined (i.e, when performing target enrichment or in organisms such as bacteria with small genome sizes). OBA is also expected to increase with the use of fragmentation methods that do not shear DNA randomly (e.g., sequence-specific restriction enzymes such as those recommended by Abascal et al. [4]). OBA can cause genuine mutations to be filtered out as artifacts, thereby reducing estimates of mutation rate [3]. Thus, exogenous UMIs should be considered when working with small genomes, targeted regions of larger genomes, or when utilizing non-random fragmentation methods.

Though the theoretical error rate for Duplex Sequencing is $<10^{-9}$/bp [5], the rate has been shown to be higher in practice (10^{-4} to 10^{-7}; [6]. The higher than expected error rate is thought to stem largely from DNA damage induced during the DNA fragmentation step of standard Duplex Sequencing library preparation [6]. Specifically, fragmentation generates single strand overhangs, as well as internal nicks and gaps, and where these single stranded regions are damaged, the end repair process converts them to double-stranded errors [4,6]. Abascal et al. [4] estimated that sonication and subsequent end repair introduced ca. 1200 mutations per diploid human cell, significantly inflating estimates of mutation rate by BotSeqS [7].

Because these artifacts are primarily the result of end-repair, removing several base pairs from the ends of consensus reads has been shown to considerably reduce the number of artifacts. For example, You et al. [6] showed that the 7 terminal base pairs on each end of a consensus read have elevated error rates, and removing these bases cut the resulting estimate of mutation rate in half. We took a more conservative approach and removed the first and last 15 bp from each DCS. Thus, our data should be largely free of terminal end repair artifacts. Alternative approaches include the use of single strand-specific nucleases to blunt 3' overhangs [8] or using blunt-end restriction enzyme-based fragmentation [4], eliminating the need for end repair.

However, fragmentation also produces internal nicks and gaps, so eliminating or computationally removing 5' ends of consensus sequences does not remove all artifacts [4,6]. Thus, additional strategies aimed at mitigating the errors associated with this type of damage should reduce error rates and improve overall estimates of somatic mutation rate. Different methods of DNA fragmentation such as enzymatic fragmentation, mechanical shearing, and nebulization all are associated with different spectra of errors [6,9], so one could reduce errors by excluding mutations associated with the utilized fragmentation method. In order to directly prevent end repair and nick extension errors, Abascal et al. [4] described a modified form of Duplex Sequencing, - Nanorate Sequencing or NanoSeq - which, in addition to generating blunt ended DNA fragments, also includes non-A dideoxynucleotides (ddBTPs) in the A-tailing reaction. As a consequence, most nick extension reactions will produce non-amplifiable fragments that will not be sequenced. Abascal et al. [4] estimate that NanoSeq achieves an error rate less than 5 x 10^{-9}/bp, approaching the theoretical error rate of Duplex Sequencing [5]. Thus, although restriction enzyme-based fragmentation does not give random sampling of the entire genome, NanoSeq is likely to yield more accurate estimates of the somatic mutation rate than other SMS approaches.

Bioinformatics scripts

```
Pipeline and dependencies setup
    a.  #Install miniconda
           i.   $ wget
                https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
          ii.   $ bash Miniconda3-latest-Linux-x86_64.sh
    b.  Install mamba via conda install
           i.   $ conda install -y -c conda-forge mamba
    c.  #Install all other dependencies via mamba install
           i.   $ mamba install -y -c bioconda -c conda-forge snakemake=5.25.0
          ii.   $ mamba install -y -c bioconda -c conda-forge pandas
         iii.   $ mamba install -y -c bioconda bwa
          iv.   $ mamba install -y -c bioconda -c conda-forge blast=2.6.0
           v.   $ mamba install -y -c bioconda -c conda-forge samtools
          vi.   $ mamba install -y -c bioconda -c conda-forge picard
    d.  #Download pipeline and update to latest version
           i.   $ git clone https://github.com/KennedyLabUW/Duplex-Seq-Pipeline.git
          ii.   $ git pull
         iii.   $ git checkout -f v2.0.0_prerelease
          iv.   $ rm .env_initialized
           v.   $ bash setupDS.sh [#cores]
Reference genome and bed file setup
    e.  #Index reference genome
           i.   $ bwa index IASC_sorted.fasta
          ii.   $ samtools faidx IASC_sorted.fasta
         iii.   $ picard CreateSequenceDictionary R=IASC_noNs.fasta
                O=IASC_noNs.dict
    f.  #Create and format bed file covering entire reference genome
           i.   $ bioawk -c fastx '{print $name"\t0\t"length($seq)}' input.fasta > output.bed
          ii.   $ awk '{print $0,"name"NR,".","."}' output.bed > output_6col.bed
         iii.   $ tr -s " " "\t" < output_6col.bed > genome.bed
```

Run pipeline

g. #Run pipeline. Navigate to directory below the main pipeline directory but above the directory containing the input files and run the pipeline. Config.csv should be edited to the desired parameters of the run.

   i.
```
$ bash ../DS config.csv
```

a. #Call variants

   i.
```
$cd /path/to/run_directory/Final/dcs
```

   ii.
```
$ for i in *dcs.final.bam; do bcftools mpileup -f
/vol_b/RefGenome/IASC_sorted.fasta $i | bcftools call -mv -V indels Ov -o
$i.vcf; done
```

b. #Generate mutational spectra

   i.
```
$ python helmsman.py --input /path/to/.vcf --fastafile
/vol_b/RefGenome/IASC_sorted.fasta --projectdir
/vol_b/Figures/vcfs/snps_vcf/helmsman_out
```

Filter variants

# Exclude known heterozygous sites

# Find and exclude intersection of initially called variants and known heterozygous sites

```
$ bedtools intersect -u -a XS_i_0.vcf.bed -b IA.BISNP.vcf.bed
```

# Exclude regions obtained by running RepeatMasker

# Extend windows excluded by RepeatMasker by 500 bp

```
$ bedtools slop -b 500 -i IASC.5000.fasta.out.bed -g
/vol_b/RefGenome/IASC_sorted.genome >
RepeatMasker_exlusions_slop500.bed
```

# Exclude regions where original reads used to build reference genome have over 2x expected coverage in reference genome.

# map reads used to build reference assembly back to reference

```
$ bwa mem -t 24 /vol_b/RefGenome/IASC_sorted.fasta IASC.R1.fastq
IASC.R2.fastq > IASC_remapped.sam
$ samtools view -S -b IASC_remapped_fixed.sam | samtools sort -o
IASC_original_mapped.bam -
```

# create coverage graph and get intervals with unusually high coverage

```
$ bedtools genomecov -ibam IASC_original_mapped.bam -bg >
IASC_original_remapped_bg.bed
$ awk '$4 > 80' IASC_original_remapped_bg.bed >
coverage_over80_IASC_original.bed
$ bedtools slop -b 500 -i coverage_over80_IASC_original.bed -g
/vol_b/RefGenome/IASC_sorted.genome | bedtools sort -i stdin | bedtools
merge -i stdin > coverage_over80_IASC_slop500.bed
```

# Exclude 500bp windows containing more than one variant

# divide genome into 500 base pair windows

```
$ bedtools makewindows -w 500 -s 100 -b IASC_sorted.bed >
IASC_sorted_windows_w500_s100.bed
```

# calculate variants per window and get windows with more than one variant per window

```
$ bedtools coverage -a
/vol_b/RefGenome/IASC_sorted_windows_w500_s100.bed -b
bwa_mem_all_libraries_consensuses.vcf.bed >
windows_coverage_IASC_sorted_all_libraries_vcfs.bed
$ for i in *.vcf.bed; do bedtools coverage -a
/vol_b/RefGenome/IASC_sorted_windows_w500_s100.bed -b $i | awk -F
$'\t' '$4 > 1' > $i.dense_variants.bed; done
$ bedtools slop -b 1000 -i
bwa_mem_all_libraries_consensuses.vcf.bed.dense_variants.bed -g
/vol_b/RefGenome/IASC_sorted.genome >
bwa_mem_all_libraries_consensuses.vcf.bed.dense_variants_slop1kb.be
d
```

```
# Merge filters and filter mutations
        $ cat * | cut -f -3 | bedtools sort -i stdin | bedtools merge -i stdin >
        beds_to_exclude_1.38.45.5.bed
        $ bedtools subtract -a /vol_b/RefGenome/IASC_sorted.bed -b
        beds_to_exclude_1.38.45.5.bed > IASC_sorted_excluding_1.38.45.5.bed
        $ bedtools intersect -u -a
        /vol_b/eddie/bwa_mem_all_libraries_consensuses.vcf.bed -b
        /vol_b/eddie/beds_to_be_merged/temp2/IASC_sorted_excluding_1.38.45
        .5.bed > all_libraries_vcfs_excluding_1.38.45.5.bed
```

List of dependencies:

- Miniconda (latest version)
- Mamba
- Python 3.6+
- Snakemake 5.25.*
- Pandas
- bwa 0.7.17.*
- ncbi blast 2.6.0 (only if doing decontamination)
- Samtools
- Picard
- Java
- bcftools
- bedtools

Supplementary References

1. Lee BY, Choi BS, Kim MS, Park JC, Jeong CB, Han J, et al. The genome of the freshwater water flea Daphnia magna: A potential use for freshwater molecular ecotoxicology. Aquatic Toxicology. 2019 May;210:69–84.
2. Routtu J, Hall MD, Albere B, Beisel C, Bergeron RD, Chaturvedi A, et al. An SNP-based second-generation genetic map of Daphnia magna and its application to QTL analysis of phenotypic traits. BMC Genomics. 2014 Nov 27;15(1):1033.
3. Matsumura S, Sato H, Otsubo Y, Tasaki J, Ikeda N, Morita O. Genome-wide somatic mutation analysis via Hawk-Seq^{TM} reveals mutation profiles associated with

chemical mutagens. Arch Toxicol. 2019 Sep 1;93(9):2689–701.

4. Abascal F, Harvey LMR, Mitchell E, Lawson ARJ, Lensing SV, Ellis P, et al. Somatic mutation landscapes at single-molecule resolution. Nature [Internet]. 2021 Apr 28 [cited 2021 Apr 29]; Available from: http://www.nature.com/articles/s41586-021-03477-4
5. Schmitt MW, Kennedy SR, Salk JJ, Fox EJ, Hiatt JB, Loeb LA. Detection of ultra-rare mutations by next-generation sequencing. PNAS. 2012 Sep 4;109(36):14508–13.
6. You X, Thiruppathi S, Liu W, Cao Y, Naito M, Furihata C, et al. Detection of genome-wide low-frequency mutations with Paired-End and Complementary Consensus Sequencing (PECC-Seq) revealed end-repair-derived artifacts as residual errors. Arch Toxicol. 2020 Oct 1;94(10):3475–85.
7. Hoang ML, Kinde I, Tomasetti C, McMahon KW, Rosenquist TA, Grollman AP, et al. Genome-wide quantification of rare somatic mutations in normal human tissues using massively parallel sequencing. PNAS. 2016 Aug 30;113(35):9846–51.
8. Otsubo Y, Matsumura S, Ikeda N, Yamane M. Single-strand specific nuclease enhances accuracy of error-corrected sequencing and improves rare mutation-detection sensitivity. Arch Toxicol [Internet]. 2021 Nov 12 [cited 2021 Nov 26]; Available from: https://doi.org/10.1007/s00204-021-03185-y
9. Knierim E, Lucke B, Schwarz JM, Schuelke M, Seelow D. Systematic Comparison of Three Methods for Fragmentation of Long-Range PCR Products for Next Generation Sequencing. PLOS ONE. 2011 Nov 30;6(11):e28240.
10. Ho EKH, Macrae F, Latta LC 4th, McIlroy P, Ebert D, Fields PD, et al. High and Highly Variable Spontaneous Mutation Rates in Daphnia. Molecular Biology and Evolution. 2020 Nov 1;37(11):3258–66.