# InSilicoSeq 2.0: Simulating realistic amplicon-based sequence reads

Stefan H. Lelieveld[1], Thijs Maas[1], Tessa C. X. Duk[2], Hadrien Gourlé[3] and Henk-Jan van den Ham[1]
[1] ENPICOM B.V., 's-Hertogenbosch, Netherlands
[2] Julius Center for Health Sciences and Primary Care, University Medical Center Utrecht, Utrecht University, Utrecht, the Netherlands
[3] Department of Clinical Microbiology, Umeå University, Umeå, Sweden

**Abstract**
**Motivation:** Simulating high-throughput sequencing reads that mimic empirical sequence data is of major importance for designing and validating sequencing experiments, as well as for benchmarking bioinformatic workflows and tools.
**Results:** Here, we present InSilicoSeq 2.0, a software package fully written in Python that is able to simulate realistic Illumina-like sequencing reads for a variety of sequencing machines and assay types. In addition to the existing functionality, InSilicoSeq now supports amplicon-based sequencing and comes with premade error models of various quality levels for MiSeq, HiSeq, NovaSeq and NextSeq sequencing platforms. It provides the flexibility to generate custom error models for any short-read sequencing platform from a BAM-file. Furthermore, it has improved computational performance with a reduced memory footprint compared to the original implementation. We demonstrated the novel amplicon sequencing capability by simulating Adaptive Immune Receptor Repertoire (AIRR) reads and show that simulated reads closely resemble the PHRED-scores of actual sequencing data. InSilicoSeq 2.0 generated 15 million amplicon based paired-end reads in under an hour at a total cost of €4.3e$^{-05}$ per million bases advocating for testing experimental designs through simulations prior to actual sequencing.
**Availability:** Source-code is freely available under the MIT licence at https://github.com/HadrienG/InSilicoSeq

**Introduction:**
Amplicon-based sequencing is considered the *de facto* standard in the characterization of specific target regions of a genome. Examples are the targeted sequencing of genetic markers such as the 16S rRNA gene for bacteria[1], the internal transcribed spacer (ITS) region for fungi[2], and the repertoire of the adaptive immune receptor repertoire (AIRR) of B and T cells[3,4]. However, generating experimental sequencing data can be time-consuming and resource intensive. Therefore, *in silico* simulations of amplicon sequenced data are an important tool to test and validate sequencing experiments as well as to benchmark downstream workflows and tools.

Simulating amplicon-based next generation sequencing (NGS) experiments can be utilized to test a variety of parameters to optimize sequencing strategies as well as the performance of bioinformatics tools and pipelines[5]. This includes evaluating the effects of varying sequencing depth, assessing the impact of different error rates, and studying the impact of sequencing technologies[6]. Thus, the insights gained from running simulated data guides the scientist to make an informed decision on what approach will give the desired results.

Here, we present an updated and extended version of InSilicoSeq, v2.0, a Python-based read simulation package that contains many new features, including amplicon-based sequencing

and new error models for the latest Illumina sequencing platforms. Existing features have been improved resulting in a constant memory-footprint, corrected read-sampling and fragment lengths, and high-resolution logging of the sequencing errors. We demonstrate the novel and easy to use amplicon sequencing capability of InSilicoSeq 2.0 by AIRR sequencing data of paired-end 301 base-pair long reads. The software-package is freely available on https://github.com/HadrienG/InSilicoSeq.

## 2. Features and benchmarks

### 2.1 Features

As described in the original publication, InSilicoSeq can simulate metagenomic data; can accurately model PHRED scores based on real sequencing data; supports substitution, insertion, and deletion errors; and supports setting of insert size distribution and GC bias by implementing Kernel Density Estimation (KDE)[7]. In line with simulating amplicon-based sequencing data (Supplementary figure 1), new error models based on the Illumina MiSeq and NextSeq sequencing platforms (paired-end reads of 301bp) have been precomputed and included in the software package (See supplementary table 1 and supplementary figure 2 for a complete overview). For the MiSeq platform, a total of five error models have been generated each based on reads with a different average PHRED-score (20, 24, 28, 32, and 36; figure 1F-H and supplementary figure 2). This allows simulating sequencing reads with a wide variety of qualities. The NextSeq reads showed high quality scores and therefore only a single error model was constructed for this platform (Figure 1D-E). The MiSeq error models and the NextSeq model are based on the data of all eight SRA entries in project PRJEB20178[8] and Illumina base space demo data project 370456276[9], respectively.

The type of sequencing data simulated by InSilicoSeq 2.0 can be controlled by the novel "—sequence_type" parameter. The average fragment length of the paired-end reads can now be specified by the user via the "—fragment-length" and "—fragment-length-sd" parameters. The InSilicoSeq amplicon simulating mode requires a fasta-formatted file containing a library of amplicon sequences in combination with a read count file that contains, for each amplicon in the library file, the number of reads that will be generated (Figure 1A; example input files can be found as supplementary data 1-3). Finally, a precomputed error model can be selected to start the amplicon read simulation (Supplementary table 1). In addition, InSilicoSeq 2.0 provides the flexibility for the user to generate custom error models from BAM-files. The output of InSilicoSeq 2.0 contains the forward (R1), and reverse (R2) reads as separate fastq-files and for each read the sequencing errors are stored in a separate vcf-formatted file (Figure 1A; supplementary data 4) with the purpose of being used as ground-truth data for detecting and correcting sequencing errors.

### 2.2 Benchmarks

In an independent and recently published article, the performance of six popular short-read simulators: ART[10], DWGSIM[11], Mason[12] NEAT[13], wgsim[14], and InSilicoSeq (version 1.5.4) were assessed[15]. The authors evaluated the read simulators on their ability to accurately and faithfully mimic characteristic features of real data obtained from Illumina sequencers. The comparison highlighted InSilicoSeq's ability to build custom error models based on empirical data and that it offers one of the largest ranges of build-in platform specific error models. Furthermore, the benchmark in the manuscript revealed some shortcomings of InSilicoSeq namely: 1) Oversampling of reads, 2) scalability, and 3) fragment lengths that were distinctly

different from those observed in the dataset. Encouraged by the outcome of this benchmark, these shortcomings have been addressed in InSilicoSeq version 2.0 (Figure 1B, 1C, supplementary figure 1 and supplementary table 3; Details of the memory and wall-time usage benchmark are available as supplementary data).

**2.3 Usability**

A benchmark was performed between InSilicoSeq 2.0 and InSilicoSeq 1.5.4 to demonstrate the improvements made in InSilicoSeq 2.0, in particular in scalability (Figure 1B and 1C) as well as simulating the correct number of reads with accurate fragment lengths (Supplementary table 3 supplementary figure 1). The benchmark of InSilicoSeq 2.0 was carried out on a Scaleway PRO-XS and PRO2-M cloud-based instance utilizing 4CPU's and 16CPU's respectively[16]. Based on a library file that mimics an AIRR sequencing experiment based on Takara's SMARTer Human BCR kit-like (Takara Bio USA) amplicons. The benchmark of InSilicoSeq, version 1.5.4 was carried out on a Scaleway PRO2-M cloud-based instance using 4CPU's or 16 CPU's. To avoid memory issues during the InSilicoSeq 1.5.4 simulations, a larger and more costly PRO2-M instance was used and downgraded to use 4 CPUs. Performance was based on simulating paired-end MiSeq based reads with a length of 301bp of the referenced assembly for Baker's yeast strain s288C (sacCer3; accession number: GCA_000146045) as described in Millhaven *et al.*[15]

The benchmark revealed InSilicoSeq 2.0 now simulates the correct number of reads with accurate fragment sizes. Furthermore, the benchmark demonstrated improved scalability with consistent memory usage, independent of the simulation size (Figure 1B and 1C, supplementary table 2 and supplementary figure 1). A direct comparison of the average per-base quality-score (PHRED) of the simulated AIRR dataset to empirical sequencing data revealed that the quality of the reads simulated by InSilicoSeq 2.0 closely resemble the empirical sequencing data (Figure 1D-H). Finally, based on the monetary costs of operating the cloud-based instances, it would require within the range of €$4.13E^{-05}$-€0.00018 to simulate 1 million bases using InSilicoSeq 2.0 (Supplementary table 2). Considering the estimated expense of $0.006 to sequence one million bases[17], simulating an equivalent number of bases would require at most 1/100$^{th}$ of the cost advocating for evaluating experimental designs through simulations prior to actual sequencing.

**3. Conclusion**

InSilicoSeq 2.0 has been designed to offer user-friendliness to both bioinformaticians and wet-lab scientists and can be executed on a wide range of computing systems. We demonstrated the novel amplicon sequencing feature of InSilicoSeq 2.0 by creating simulations of paired-end reads of Adaptive Immune Receptor Repertoire (AIRR) data with a length of 301 base-pairs. In addition, we have added the 301 base-pair paired error model based on Illumina NextSeq data, as well as error models based on Illumina MiSeq data, considering various averaged PHREDs-scores to simulate amplicon reads across several quality-classes. We have shown, based on the AIRR sequencing example, that the software package produces realistic amplicon-based sequencing reads that closely resembles the PHRED-scores of empirical reads. The performance assessment of InSilicoSeq 2.0 showed that the performance shortcoming reported in a recent and published benchmark study have been addressed, resulting in a scalable tool with constant memory usage regardless of the simulation size: Furthermore, the use of cloud-based instances enabled to estimate the

monetary expenses to simulate 1M bases at a cost of €4.13E-05-€0.00018. This makes InSilicoSeq 2.0 an inexpensive approach for validating sequencing experiments as well as bioinformatic workflows and allows scientists to make an informed decisions on what settings will give the desired results.

**Availability**

The software-package is freely available on https://github.com/HadrienG/InSilicoSeq under the MIT licence.

**Funding**

This project has received funding from the Innovative Medicines Initiative 2 Joint Undertaking under grant agreement No 101007799 (Inno4Vac). This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and EFPIA. This communication reflects the author's view and neither IMI nor the European Union, EFPIA, or any Associated Partners are responsible for any use that may be made of the information contained therein. HG was supported by the SciLifeLab & Wallenberg Data Driven Life Science Program (grant: KAW 2020.0239).

**Conflict of interest**

The authors declare no conflict of interest.

**Acknowledgements**

The computations were also partially enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) and the Swedish National Infrastructure for Computing (SNIC) at UPPMAX partially funded by the Swedish Research Council through grant agreements no. 2022-06725 and no. 2018-05973.

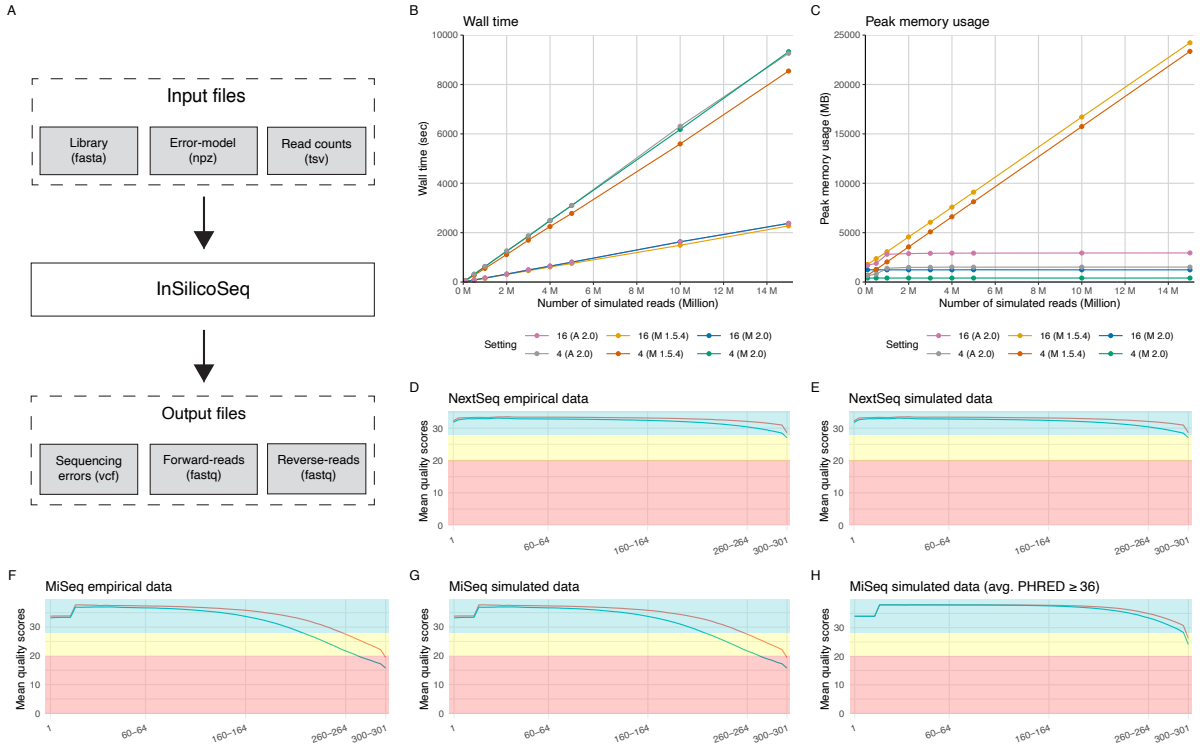
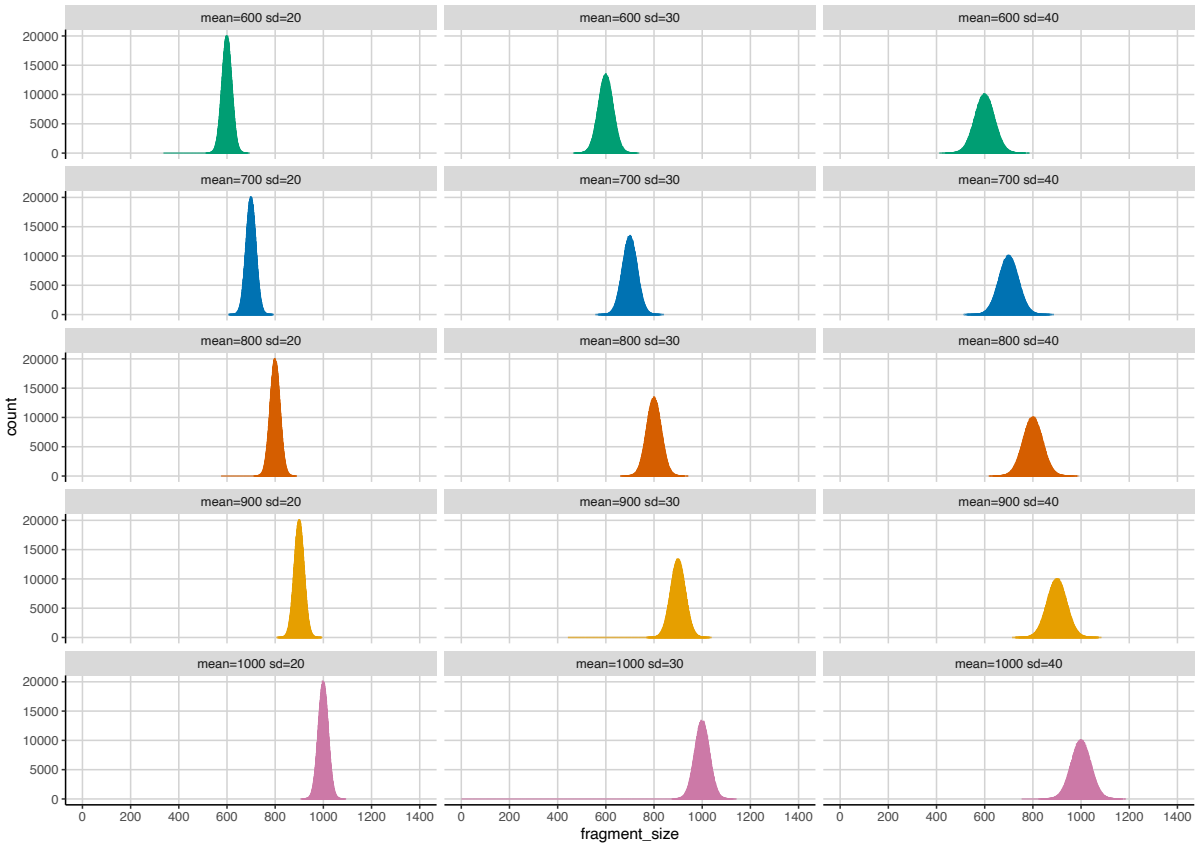
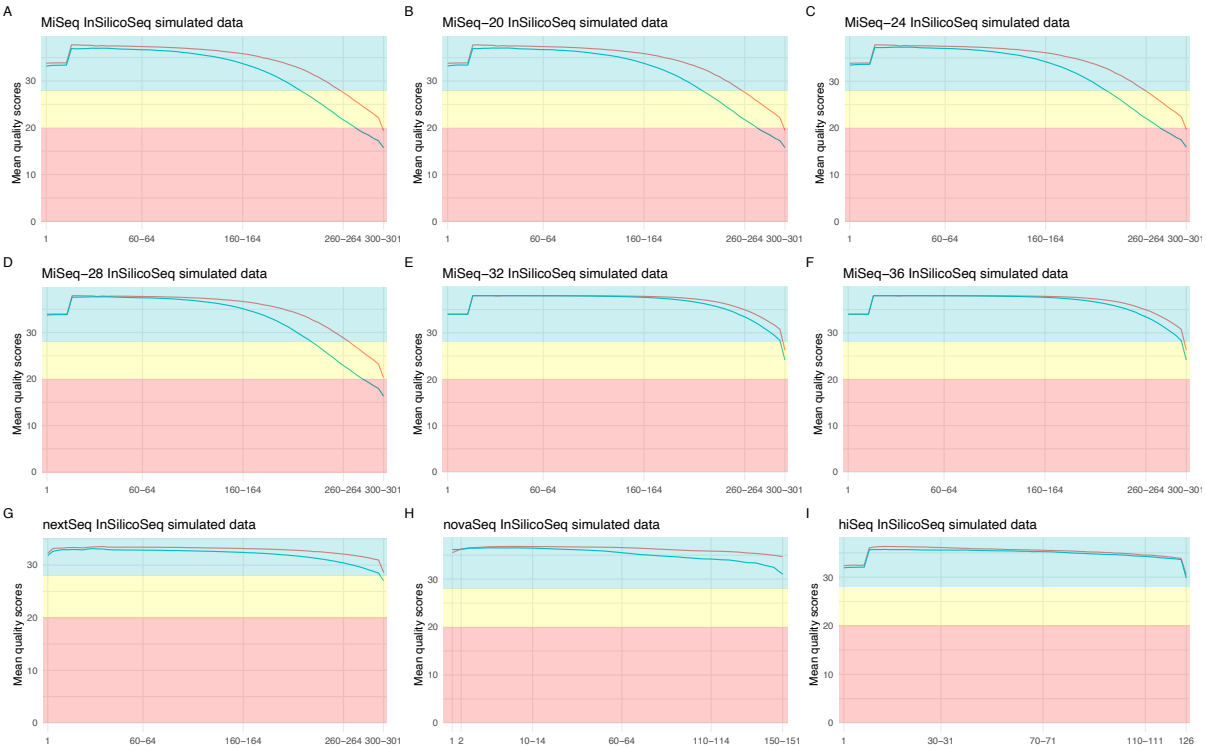



**Figure 1**. (**A**) Overview of the workflow to generate amplicon-based reads with InSilicoSeq 2.0. Benchmarking (**B**) Wall time and (**C**) Peak memory usage for amplicon and meta genome simulations executed by InSilicoSeq 2.0 and 1.5.4 with various number of reads performed on a cloud-based instance with 4 CPU's (version 1.5.4 metagenome-mode in orange, version 2.0 metagenome-mode in green and amplicon-mode in grey ) and 16 CPU's (version 1.5.4 metagenome-mode in yellow and version 2.0 metagenome-mode in blue and amplicon-mode in pink). The Python module "memory_profiler" was used to determine the wall time and peak memory usage of each run[18]. Mean quality scores (PHRED) per base of the forward (red line) and reverse (green line) 301base-pair reads based on: (**D**) empirical Illumina NextSeq data (Illumina base space project 370456276) and (**E**) simulations based on the empirical NextSeq data. Idem for (**F**) Empirical Illumina MiSeq data of SRA project PRJEB20178, (**G**) a simulation based on the empirical MiSeq data and (**H**) a simulation based on an error model of filtered empirical MiSeq reads with an average score of at least 36. The mean quality score plots have been generated using the "fastqcr" R-package.[19,20]

## SUPPLEMENTARY FIGURES




**Supplementary Figure 1:** Overview of metagenome simulations based on the baker's yeast genome with various fragment length. InSilicoSeq 2.0 have two new parameters to control the average length of the simulated fragments (--fragment-length) and the standard deviation (--fragment-length-sd). The bar plots show the distribution of fragment lengths for different combinations of average fragment length values (600, 700, 800, 900, 1000) and standard deviation values (20, 30, 40).




**Supplementary figure 2:** Overview of the per-base quality (PHRED) scores for each precomputed error model that is available through the InSilicoSeq 2.0 cli. (**A**) default MiSeq error model with paired end reads of 301bp. MiSeq error models with paired end reads of 301bp and the forward and reverse reads have (**B**) an average quality score of at least 20 (**C**) an average quality score of at least 24 (**D**) an average quality score of at least 28 (**E**) an average quality score of at least 32 (**F**) an average quality score of at least 36. (**G**) NextSeq error model with paired end reads of 301bp (**H**) NovaSeq error model with paired end reads of 151bp (**I**) HiSeq error model with paired end reads of 126bp

**SUPPLEMENTARY TABLES**

| Error model | Sequencing platform | Read length | Filters | Cli |
|---|---|---|---|---|
| nextSeq.npz | NextSeq | 2x 301bp | None | nextseq |
| miSeq_20.npz | MiSeq | 2x 301bp | Average PHRED score ≥ 20 | miseq-20 |
| miSeq_24.npz | MiSeq | 2x 301bp | Average PHRED score ≥ 24 | miseq-24 |
| miSeq_28.npz | MiSeq | 2x 301bp | Average PHRED score ≥ 28 | miseq-28 |
| miSeq_32.npz | MiSeq | 2x 301bp | Average PHRED score ≥ 32 | miseq-32 |
| miSeq_36.npz | MiSeq | 2x 301bp | Average PHRED score ≥ 36 | miseq-36 |
| miSeq.npz | MiSeq | 2x 301bp | None | miseq |
| hiSeq.npz | HiSeq | 2x125bp | None | hiseq |
| novaSeq.npz | NovaSeq | 2x150bp | None | novaseq |

**Supplementary table 1:** Overview of the precomputed error models for different Illumina sequencing platforms that are available through the InSilicoSeq 2.0 cli.

**Supplementary_table_2.xlsx :** Performance benchmark of InSilicoSeq simulating AIRR sequencing based experiments with 301bp paired-end reads. Simulations with 4 CPUs were performed on a PRO2-XS Scaleway instance (4 cores, 16Gb RAM, 100Gb block SSD, Ubuntu 22.04, with an estimated cost of €0.1258/Hour). Simulations with 16 CPUs were performed on a PRO2-M Scaleway instance (16 cores, 64Gb RAM, 100Gb block SSD, Ubuntu 22.04, with an estimated cost of €0.4538/Hour)

**Supplementary_table_3.xlsx :** Comparison of the expected number of reads and the number of simulated reads shows a perfect match.

**SUPPLEMENTARY DATA-FILES**

**Supplementary File 1: library.fasta**

**Supplementary File 2: read_count.tsv**

**Supplementary File 3: sequencing_errors.vcf**

**Supplementary File 4: miseq_10K_reads_R1.fastq**

**Supplementary File 5: miseq_10K_reads_R2.fastq**

**Supplementary File 6: run_iss.sh**

## References


1. Methé, B. A. *et al.* A framework for human microbiome research. *Nature* **486**, 215–221 (2012).

2. Schoch, C. L. *et al.* Nuclear ribosomal internal transcribed spacer (ITS) region as a universal DNA barcode marker for Fungi. *Proc. Natl. Acad. Sci.* **109**, 6241–6246 (2012).

3. Robins, H. S. *et al.* Comprehensive assessment of T-cell receptor β-chain diversity in αβ T cells. *Blood* **114**, 4099–4107 (2009).

4. Wu, X. *et al.* Focused Evolution of HIV-1 Neutralizing Antibodies Revealed by Structures and Deep Sequencing. *Science* **333**, 1593–1602 (2011).

5. Lind, A. L. & Pollard, K. S. Accurate and sensitive detection of microbial eukaryotes from whole metagenome shotgun sequencing. *Microbiome* **9**, 58 (2021).

6. Andreu-Sánchez, S. *et al.* A Benchmark of Genetic Variant Calling Pipelines Using Metagenomic Short-Read Sequencing. *Front. Genet.* **12**, 648229 (2021).

7. Gourlé, H., Karlsson-Lindsjö, O., Hayer, J. & Bongcam-Rudloff, E. Simulating Illumina metagenomic data with InSilicoSeq. *Bioinformatics* **35**, 521–522 (2018).

8. LANTBRUKSUNIVERSITET, S. PRJEB20178. https://www.ncbi.nlm.nih.gov/bioproject/PRJEB20178 (2018).

9. Illumina base space demo files. https://basespace.illumina.com/datacentral.

10. Huang, W., Li, L., Myers, J. R. & Marth, G. T. ART: a next-generation sequencing read simulator. *Bioinformatics* **28**, 593–594 (2012).

11. DWGSIM. https://github.com/nh13/DWGSIM.

12. Holtgrewe, M. Mason: a read simulator for second-generation sequencing data. *Technical Report FU Berlin* (2011).

13. Stephens, Z. D. *et al.* Simulating Next-Generation Sequencing Datasets from Empirical Mutation and Sequencing Models. *PLoS ONE* **11**, e0167047 (2016).

14. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078–2079 (2009).

15. Milhaven, M. & Pfeifer, S. P. Performance evaluation of six popular short-read simulators. *Heredity* **130**, 55–63 (2023).

16. Scaleway Instance pricing. https://www.scaleway.com/en/pricing/?tags=compute.

17. Wetterstrand, K. A. DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP). www.genome.gov/sequencingcostsdata.

18. Pedregosa, F. & Gervais, P. memory profiler.
https://github.com/pythonprofilers/memory_profiler.

19. Andrews, S. A quality control tool for high throughput sequence data.
https://www.bioinformatics.babraham.ac.uk/projects/fastqc/ (2010).

20. Kassambara, A. fastqcr: Quality Control of Sequencing Data.
https://rpkgs.datanovia.com/fastqcr/index.html (2023).