ISSN 1995-0802, Lobachevskii Journal of Mathematics, 2022, Vol. 43, No. 8, pp. 2092–2098. © Pleiades Publishing, Ltd., 2022.

Deep Learning for Inferring Distribution of Time to the Last Common Ancestor from a Diploid Genome

K. Arzymatov[1*], **E. Khomutov**[1**], **and V. Shchur**[1***]

(Submitted by A. M. Elizarov)

[1] *National Research University Higher School of Economics, Moscow, 101000 Russia*

Received April 22, 2022; revised May 6, 2022; accepted May 14, 2022

Abstract—Genomic data is a rich source of information about population history. In particular, for actively recombining species the time to the last common ancestor (LCA) between two chromosomes might be different in different chromosome loci. Estimating local LCA time is important for many problems: it can be used to infer genes under selection, or to infer effective population size changes. The current state-of-the art method PSMC to infer local LCA time and effective population size is based on a Hidden Markov Model. In this work we propose a new deep learning framework for local LCA time inference at the full genome scale. We demonstrate that our method is accurate in both local LCA time and, as a consequence, at the LCA time distribution which in turn translates into effective population size trajectory. In future our approach can be generalised for complex population scenarios.

DOI: 10.1134/S1995080222110075

Keywords and phrases: *deep learning, population genomics, LCA, effective population size, demography, genome, chromosome.*

1. INTRODUCTION

Population genetics is a recent interdisciplinary research area which relies on mathematical and computational methods to infer population demography and structure through time from genomic data. Whole-genome sequencing data are rich in such information. But due to the genome length and the complexity of underlying processes, these data are challenging to be analysed.

Demographic inference, which assumes an estimation of the population size changes through time, is one of the key problems in population genetics. For example, it was shown [1] that all the non-African populations underwent through a similar bottleneck between approximately 30kya (20 thousand years ago) and 100kya. African populations do not show signals of such a bottleneck. This fact supports the hypothesis of an African origin of modern human population.

The state-of-the-art methods for demographic inference used for mammals and some other animals, are based on a Hidden Markov Model (HMM). Such an HMM decodes genetic variants along a genome under Sequential Markovian Coalascent (SMC) [2, 3]. SMC is a population model which approximates coalescent with recombination [4]. Under coalescent with recombination, a genealogy of a set of chromosomes is a directed acyclic graph with an additional data structure reflecting spacial structure along these chromosomes. Limiting this genealogy to a single genetic site, results in a tree. These trees might change between adjacent genetic sites because of ancestral recombinations. Under SMC approximation, genealogical trees change along a genome following a Markovian process. Transitions between trees are caused by chromosomal recombinations. Under HMM, these genealogical trees are hidden states, and genetic variants are emissions. Different methods rely on this idea including

[*] E-mail: karzymatov@hse.ru

[**] E-mail: ehomutov@hse.ru

[***] E-mail: vshchur@hse.ru

PSMC [1], diCal [5], MSMC [6], SMC++ [7], ASMC [8], MSMC-im [9], ngsPSMC (unpublished). These methods differ in the structure of input data (a single diploid genome for PSMC, multiple phased genomes for MSMC, a single diploid sample with allele frequencies of genetic variants for SMC++, sparse genotype data for ASMC, genotype likelihoods for a single diploid genome for ngsPSMC).

Deep learning is also used in population genetics for studying demographic, population and evolutionary parameters, though these methods are not widely used. A detailed review of deep learning methods in population genetics is provided in [10], so we discuss only few important examples here. Mondal et al. [11] used deep learning to generate a genomic summary statistics for Approximate Bayesian Computation (ABC) [12], to study archaic introgression in Asia and Oceania. In [13] deep learning is used to jointly infer demography and natural selection. In [14] exchangeable neural networks and on-the-fly simulations were used to infer recombination hotspots. A first method to infer a detailed population history with deep learning is proposed in [15] and proves that neural network based approach could be a powerful tool in population genetics. Though more efforts should be done in the field to explore the advantages and limitations of deep learning to be used for real data analyses. In this work we suggest a new deep learning framework to predict LCA times along a diploid genome at a whole genome scale, similarly to PSMC [1].

There are two important challenges in population genetics for deep learning applications. Firstly, it is a length of a genome. For example, a human genome is around 3.2 billion nucleotides, which is by several orders of magnitude longer than any text in natural language processing (NLP) problems. Secondly, it is the absence of labeled real data. This challenge is solved through simulating training datasets, benefiting from efficient simulators available in the field.

Some other complications in genome wide inference are the variable genome size, sequencing errors (and other potential sources of errors such as post-mortem damage in ancient DNA), missing data, low depth sequencing (which leads to the errors in variant calling), and some others. While our approach addresses the first complication, all other aspects are currently off the scope of this paper.

Our framework is implemented as a modular package. It is aimed to become a helpful tool for population geneticists who are not experts in neural networks to allow a fast prototyping of their own DL-based methods. The code can be found at https://github.com/Genomics-HSE/deepgen.

2. MODELS AND METHODS

2.1. Basic Biological Notations

In this paper, we use the following biological terms. *Genome* is the genetic material of an organism which consists of multiple DNA molecules (formally, a string over an alphabet of four letters, nucleotides, 'A', 'T', 'C' and 'G'). Each such DNA molecule is called a *chromosome*. Within a species, genomes of all individuals can be aligned relatively to each other. The differences between genomes relatively to such an alignment are called genetic variants and arise due to ancestral mutations, some of which might appear hundreds of thousands generations ago. A position in this alignment is called a *site*. A site with a genetic variant is a *variable site* (in fact, there are many types of genetic variability but we consider only the most common Single Nucleotide Polymorphism, shortly SNP). Genomes of diploid organisms, such as humans, normally contain two sets of chromosomes, called *haplotype* from each of two parents.

All the analysis in this paper is based on the distribution of pairwise differences between two haplotypes along the genome. So, this data can be encoded as a binary sequence with 0 referring to identical nucleotides at the site, *homozygous* site, and 1 referring to a genetic variant between these two haplotypes, *heterozygous* site.

Effective population size is one of the key quantities in population genetics [16]. It can be defined as a population size of Wright—Fisher population [17] with the same genetic diversity. Another informal interpretation of this quantity is the number of individuals in a population which potentially can have offsprings. This interpretation underlines an important differences between effective population size and the census size (the total number of individuals in a population).

For the Wright—Fisher model [17], effective population size uniquely defines the distribution of the time to the last common ancestor (LCA). As it is shown in [1], this distribution can be estimated from a single diploid genome in case of a recombining organism. Due to recombinations, every chromosome is a mosaic of DNA tracts coming from different ancestors. So, the time to the last common ancestor between two haplotypes varies along the genome. Tracts with deep ancestry would be short and at the same time have relatively many pairwise differences which are due to ancestral mutations. On the opposite, tracts with recent ancestry tend to be longer and have a small amount of heterozygous sites.

2.2. Method Overview

As we explained above, the input data for our method is a binary sequence encoding pairwise differences between two haplotypes with 0 for homozygous sites and 1 for heterozygous sites. The target is the time to the local last common ancestor along each chromosome. We state this problem as classification approach. The time axis is divided into multiple time intervals. Similarly to PSMC, the length of these time intervals increases exponentially while going deeper in the past. More precisely, we first choose some $T_{\max}$—the last non-infinite end-point of the time axis partition. Then we choose the number K of time intervals in the partition, and we set end-points T_k

$$T_k = \alpha(e^{k \log(1+10T_{\max})/K} - 1).$$

Because there is a very small chance for two lineages to coalesce within the first few time intervals, we merge the first 4 time intervals into a single one.

In all the following results we use the following parameter values $T_{\max} \approx 95000$ generations, $K = 32$, $\alpha = 550$.

2.3. Training Dataset

In the absence of labeled real data with known time to the last common ancestor, we generate a training dataset through simulations similar to [13]. Mutation and recombination rates were fixed with human-like values of mutation rate $\mu = 1.25 \times 10^{-8}$ and recombination rate $\rho = 1.6 \times 10^{-9}$ [9, 15].

We present results for two training datasets. The first dataset (labeled "dataset 1") consists of chromosomes sampled from a demography with a fixed constant effective population size of 1 (in the coalescent units of $2N_0$).

The second dataset (labeled "dataset 2") consists of randomly drawn demographies. The time axis was split into 13 time intervals, with the effective population size being 1 at the last time interval, and for the rest of 12 time intervals effective population sizes were randomly drawn. A bottleneck was allowed during the 4th, 5th, and 6th time intervals. The code for generating the trajectories is available at the GitHub repository associated with the project.

2.4. Data Preparation

The dataset consists of learning pairs, with the first element of a pair encoding a diploid genome, and the second element represents labels (local LCA times). A genome is encoded with 0's (homozygous sites) and 1's (heterozygous sites). In order to reduce the size of each example (in particular, due to GPU memory limitations), we split chromosomes into segments of 30 million base pairs. Further, nucleotides are binned into 100bp non-overlapping windows. If all positions within a window are homozygous, the window is encoded by 0, otherwise, it is encoded by 1. The same procedure is used in PSMC [1] analysis. As a result, we use sequences being 30000 bin long for training.

2.5. Architecture

Recurrent neural networks proved to be a powerful approach for sequence data analysis. Our model uses 4 layers of bidirectional GRU network [18]. Bidirectional networks allow to combine information both from left and right from each position on a sequence with non-shareable set of weighs for each of the two (left-to-right and right-to-left) RNNs. Input of each layer of GRU network is an output of the previous layer, and it also processes these inputs in a bidirectional manner. The last building block of NN are fully-connected layers which make a final label prediction for a locus. Resulting model is represented schematically on Fig. 1.

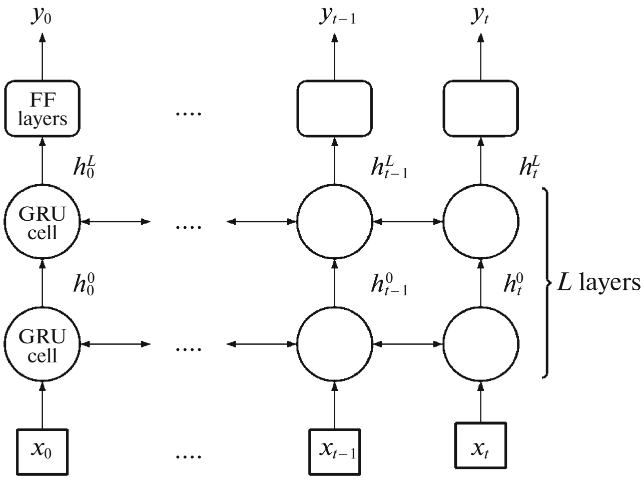

Fig. 1. Neural network architecture. x_t are genome positions (or bins) being in homozygous (0) or heterozygous (1) states. h_t^l is a hidden vector with a relevant information about current locus, FF are fully connected layers and y_t is a final prediction of the model (a vector of probabilities that local LCA is in a certain time interval).

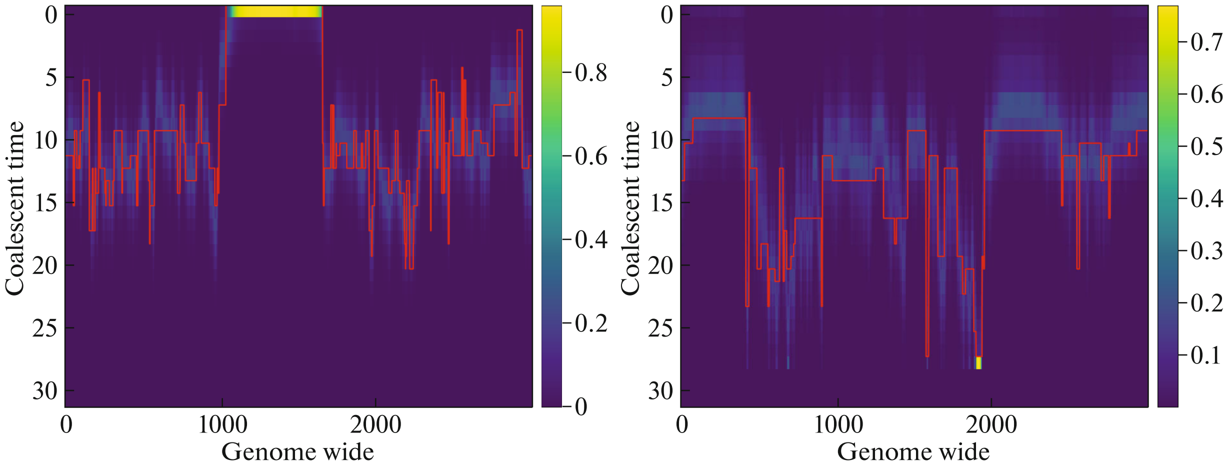

Fig. 2. The time of LCA along a chromosome. The heatmaps show the probabilities of LCA time being at a time interval along a chromosome. Left panel demonstrates the LCA time inferred from a chromosome representing a constant size population by the neural network trained on a constant size dataset. Right panel represents LCA time along a chromosome from a population with a bottleneck, the neural network was trained on a dataset with random demographies. Red line represents the true underlying LCA times (known from simulations).

3. RESULTS

As explained in the Methods section, we trained our deep learning model on two different training datasets. We applied these neural networks to two diploid genomes: the first genome (labeled “const”) is drawn from a demographic scenario with constants effective population size; the second genome (labeled “bottleneck”) is drawn from a demographic scenario with a human out-of-Africa like bottleneck (represented in `ms` [19] syntax as `-eN 0.0 3.0 -eN 0.025 0.2 -eN 0.175 1.5 -eN 3 3 -eN 10.0 3`).

We present the results of the local LCA time prediction of a model trained on dataset 1 on a “const” chromosome (Fig. 2, left panel), and a model trained on dataset 2 on a “bottleneck” chromosome. Heatmap represents the probabilities of each LCA time class for each genomic position. Qualitative assessment shows that in both cases our neural network captures the LCA time rather well.

Further on, we calculated the cumulative distribution of LCA time (sum over all genomic positions) from simulated genomes, and from the neural network prediction. We investigated prediction of the dataset 1 and dataset 2 models both on “const” and “bottleneck” genomes. The results are represented

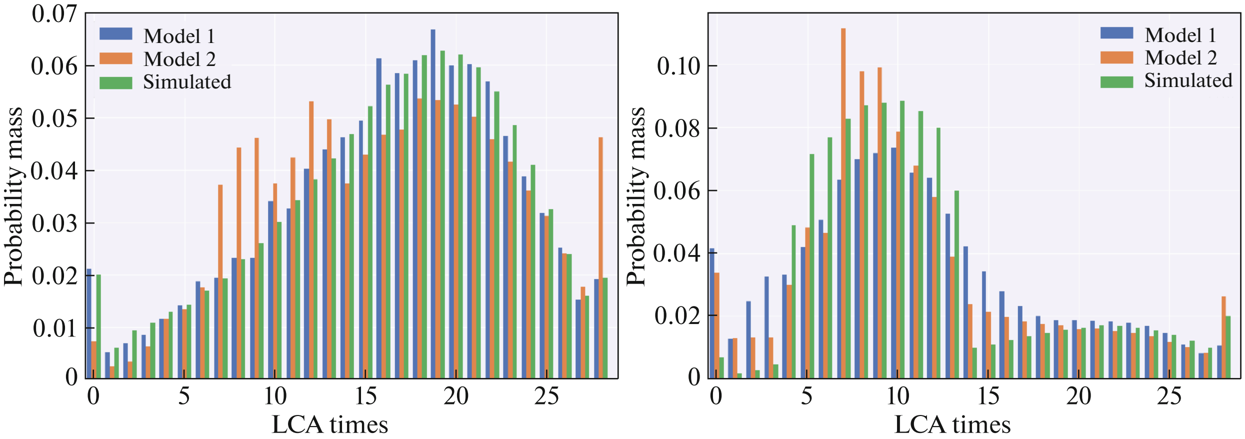

Fig. 3. The cumulative distributions of LCA times calculated from the simulated genome ("simulated"), estimated by model trained on dataset 1 ("model 1"), and by model trained on dataset 2 ("model 2").

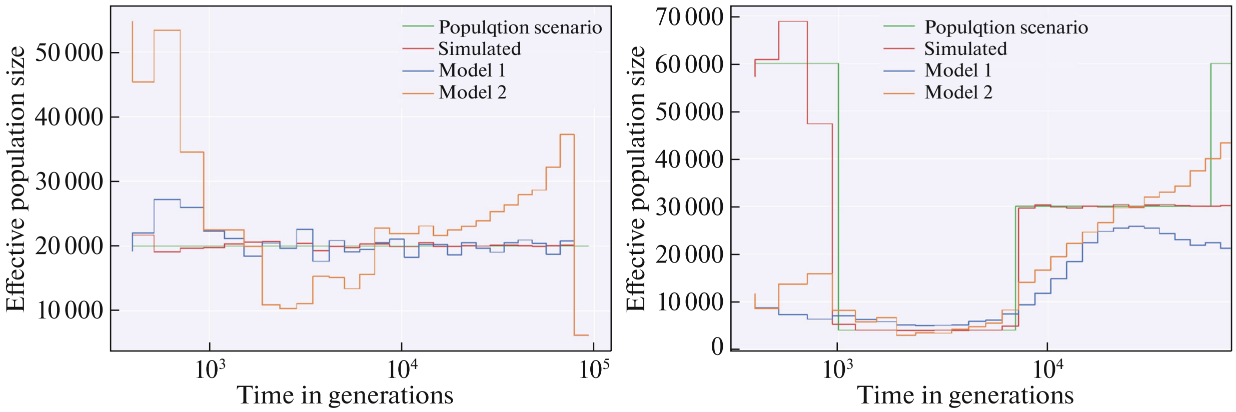

Fig. 4. Effective population size estimated as the inverse of coalescent rates calculated from the simulated genome ("simulated"), estimated by model trained on dataset 1 ("model 1"), and by model trained on dataset 2 ("model 2"). The effective population size of the simulated demographic scenario is also shown ("population scenario").

on Fig. 3. Model trained on dataset 1 performs well on the "const" genome, and model trained on dataset 2 performs well on the "bottleneck" genome. Predictions of dataset 1 model on the "bottleneck" genome and of dataset 2 model on the "const" genome are less precise. While in the first case this is not surprising (the demographic scenario is fixed, and only the chromosomes are randomly drawn from it), it is less explainable in the second case (where demographic scenarios are randomly chosen). These examples show that the prediction might be sensitive to the training dataset, and a further research is needed to understand how to generate an optimal training dataset. In fact, there are three potential sources of signal for the neural network. The first one is the density of heterozygotes in a genomic region: the larger is the LCA time, the more heterozygotes present in a region. Secondly, longer regions correspond to shorter LCA times. Third source of information is the patterns of transitions between different LCA times. In terms of a Hidden Markov Model, the first point is determined by emission matrix, while the second and the third are determined by transition matrix. Moreover, they contain information about additional population structure which will be a subject of our future work.

Cumulative LCA time distribution determines the effective population size which is represented at Fig. 4.

We also present confusion matrices (Fig. 5). These matrices represent the probabilities to predict class X given that the true class is Y. For the recent times there is a bias in both models to underestimate the LCA time. For other classes both models are rather precise which is supported by the diagonal shape of the confusion matrices.

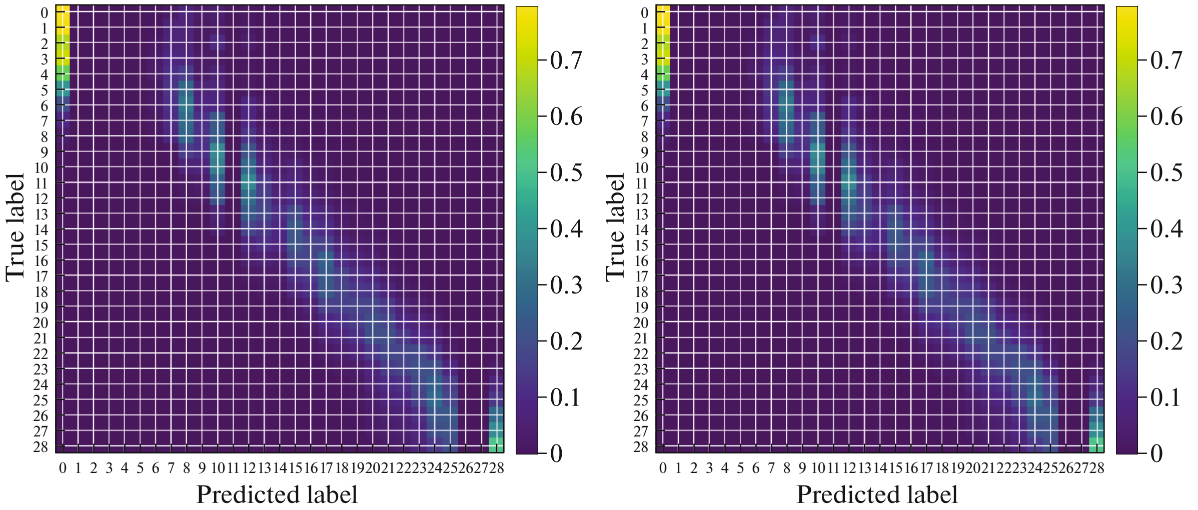

Fig. 5. Confusion matrices for models trained on dataset 1 (left) and dataset 2 (right). Each line corresponds to a probability for a model to predict class X given that the true class is Y.

3.1. Code

The code is available at GitHub repository https://github.com/Genomics-HSE/deegen.

FUNDING

E.K. and V.S. were supported by the Russian Science Foundation under grant 20-71-00143. Deep learning models were trained using HPC facilities at HSE University.

REFERENCES

1. H. Li and R. Durbin, "Inference of human population history from individual whole-genome sequences," Nature (London, U.K.) **475**, 493–496 (2011).
2. G. A. T. McVean and N. J. Cardin, "Approximating the coalescent with recombination," Philos. Trans. R. Soc. London, Ser. B **360**, 1387–1393 (2005).
3. P. Marjoram and J. D. Wall, "Fast 'coalescent' simulation," BMC Genetics **7** (2006).
4. R. R. Hudson, "Gene genealogies and the coalescent process," Oxford Surv. Evolut. **7**, 1–44 (1990).
5. S. Sheehan, K. Harris, and Y. S. Song, "Estimating variable effective population sizes from multiple genomes: A sequentially Markov conditional sampling distribution approach," Genetics **194**, 647–662 (2013).
6. S. Schiffels and R. Durbin, "Inferring human population size and separation history from multiple genome sequences," Nat. Genet. **46**, 919–925 (2014).
7. J. Terhorst, J. A. Kamm, and Y. S. Song, "Robust and scalable inference of population history from hundreds of unphased whole genomes," Nat. Genet. **49**, 303–309 (2017).
8. P. F. Palamara, J. Terhorst, Y. S. Song, and A. L. Price, "High-throughput inference of pairwise coalescence times identifies signals of selection and enriched disease heritability," Nat. Genet. **50**, 1311–1317 (2018).
9. K. Wang, I. Mathieson, J. O Connell, and S. Schiffels, "Tracking human population structure through time from whole genome sequences," PLOS Genetics **16**, 1–24 (2020).
10. G. Eraslan, Z. Avsec, J. Gagneur, and F. J. Theis, "Deep learning: New computational modelling techniques for genomics," Nat. Rev. Genet. **20**, 389–403 (2019).
11. M. Mondal, J. Bertranpetit, and O. Lao, "Approximate bayesian computation with deep learning supports a third archaic introgression in Asia and Oceania," Nat. Commun. **10** (2019).
12. S. Tavaré, D. J. Balding, R. C. Griffiths, and P. Donnelly, "Coalescence times from DNA sequence data," Genetics **145**, 505–518 (1997).
13. S. Sheehan and Y. S. Song, "Deep learning for population genetic inference," PLOS Comput. Biol. **12**, 1–28 (2016).
14. J. Chan, V. Perrone, J. P. Spence, P. A. Jenkins, S. Mathieson, and Y. S. Song, "A likelihood-free inference framework for population genetic data using exchangeable neural networks," Adv. Neural Inform. Process. Syst. **31**, 8594–8605 (2018).

15. T. Sanchez, J. Cury, G. Charpiat, and F. Jay, “Deep learning for population size history inference: Design, comparison and combination with approximate bayesian computation,” Mol. Ecol. Resour. **21**, 2645–2660 (2021).
16. P. Sjödin, I. Kaj, S. Krone, M. Lascoux, and M. Nordborg, “On the meaning and existence of an effective population size,” Genetics **169**, 1943–2631 (2005).
17. S. Wright, “Evolution in mendelian populations,” Genetics **16**, 97–159 (1931).
18. K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” arXiv: 1409.1259 (2014).
19. R. R. Hudson, “Generating samples under a Wright-Fisher neutral model of genetic variation,” Bioinformatics **18**, 337–338 (2002).