`diplo-locus`: A lightweight toolkit for inference and simulation of time-series genetic data under general diploid selection

Xiaoheng Cheng*[1] and Matthias Steinrücken†[1,2]

[1]Department of Ecology and Evolution, University of Chicago, Chicago, IL, USA.
[2]Department of Human Genetics, University of Chicago, Chicago IL,USA.

October 12, 2023

Abstract

Summary: Whole-genome time-series allele frequency data are becoming more prevalent as ancient DNA (aDNA) sequences and data from evolve-and-resequence (E&R) experiments are generated at a rapid pace. Such data presents unprecedented opportunities to elucidate the dynamics of adaptative genetic variation. However, despite many methods to infer parameters of selection models from allele frequency trajectories available in the literature, few provide user-friendly implementations for large-scale empirical applications. Here, we present `diplo-locus`, an open-source Python package that provides functionality to simulate and perform inference from time-series under the Wright-Fisher diffusion with general diploid selection. The package includes Python modules as well as command-line tools.
Availability: Python package and command-line tool avilable at: `https://github.com/steinrue/diplo_locus` or `https://pypi.org/project/diplo-locus/`.

1 Introduction

With the rapid advancements in sequencing technologies, large datasets of temporally-stratified population genomic samples are increasingly common. On the one hand, improved techniques for processing ancient DNA (aDNA) enable the collection of aDNA sequencing data for many samples in humans and other species. On the other hand, for evolution experiments of organisms in the laboratory, it has become increasingly affordable to sequence the genomes of samples taken at multiple time points, bringing forth a fast-growing body of evolve-and-resequence (E&R) datasets. These datasets enable observing changes in the genetic composition of a population across multiple time points, allowing detailed inference about the underlying processes.

One of the most intriguing forces that shape the dynamics of genetic variants is natural or artificial selection, as it reflects how the population adapts to its environment. The Wright-Fisher model and its diffusion approximations are commonly used to model the dynamics of allele frequencies over time, especially when selection is acting, and they have been applied to analyze temporal genetic data (*e.g.*, Bollback et al., 2008; Malaspinas et al., 2012; Mathieson and McVean, 2013; Steinrücken et al., 2014; Steinrücken et al., 2016; He et al., 2020). Nevertheless, despite several existing approaches to infer selection parameters, most analyze the data exclusively under a model of additive (semi-dominant) selection, and are often applied to time-series at specific loci, rather than genome-wide. However, some recent approaches do leverage full genomic data (Mathieson and Terhorst, 2022; Whitehouse and Schrider, 2023).

*xhcheng@uchicago.edu
†steinrue@uchicago.edu

Here, we present our open-source software package `diplo-locus` that implements the single-locus Wright-Fisher diffusion with piece-wise constant general diploid selection in a panmictic population (see Section S1 in *Supplementary Notes*). It can be used to simulate allele frequency trajectories and corresponding time-series samples. Moreover, it can analyze such time-series samples under a Hidden Markov Model (HMM) framework (Bollback et al., 2008; Mathieson and McVean, 2013) using efficient Gaussian approximation of the allele frequency dynamics to perform likelihood-based inference of general diploid selection coefficients. With most of the computation vectorized, `diplo-locus` is able to analyze data at a large number of loci simultaneously without compromising efficiency and speed. We believe this lightweight toolkit is a useful addition to the suite of methods for the broader scientific community working with temporal genetic samples.

2 Implementation

Our software package `diplo-locus` is developed in Python3 (3.8 and above) and can be directly installed from the `PyPI` (`https://pypi.org/project/diplo-locus/`) repository. It requires the `NumPy` (Harris et al., 2020), `SciPy`, `pandas`, and `matplotlib` packages. Once installed, the respective functions can be imported from the `diplo_locus` module in Python, or the user can alternatively use the command-line interface (CLI), `DiploLocus`, to perform analyses in a Unix/Linux environment. Both the functions of the module and the CLI implementations of this program can perform simulation and likelihood computation with a wide range of customizable options. A detailed manual and tutorials can be found in the GitHub repository at `https://github.com/steinrue/diplo_locus`.

2.1 Command-Line Interface `DiploLocus` (CLI)

To compute log-likelihoods, the command `DiploLocus likelihood` reads the genotype data either from a VCF file with a matching sample information file or from a table of allele counts with sampling times. The user can use command-line arguments to designate subsets of loci and filter loci by pooled minor allele frequency. For input as VCF in particular, the user can specify the subset of samples to be considered. If sampled individuals with pseudo-haploid and individuals with diploid genotypes should be analyzed together, the program accepts VCF files that contain a mixture of haploid and diploid samples (the command-line options `--force_hap` or `--force_dip` can be used to resolve ambiguities).

In addition to the input files, the user must provide an effective population size (`--Ne`), per-generation recurrent mutation rate(s) (`--u[01|10]`), and the initial allele frequencies (`--initCond`), either as a fixed value or a probability distribution. For values of per-generation selection coefficients, the program accommodates two fitness parameterizations (see Table S1 in *Supplementary Notes*): by relative fitness of heterozygotes and homozygotes (s_{aA} and s_{AA} at bi-allelic locus with alleles a and A) or by selection and dominance coefficients, as adopted by SLiM ($s = s_{\mathrm{AA}}$ and h; Haller and Messer, 2023). The program provides options to compute log-likelihoods on a linear or geometric grid of values for each selection parameter or at a fixed value. The program also allows for selection models with selection coefficients changing over time in a piece-wise constant manner.

Besides computing on-grid log-likelihoods, the program can interpolate 1-dimensional log-likelihood surfaces and output on- and off-grid maximum likelihood estimates (MLEs) of the selection coefficients for each locus. Furthermore, because likelihood ratio statistics asymptotically follow a χ^2 distribution (Self and Liang, 1987), the program offers the choice to output corresponding p-values computed using a χ^2 distribution with 1 degree of freedom.

To simulate temporal samples and allele frequency trajectories, the command `DiploLocus simulate` allows users to simulate an arbitrary number of independent replicates under the specified population and selection parameters. The tool can simulate using a model of constant selection throughout time, as well as piece-wise constant selection coefficients. Multiple filtering options based on sample or population frequencies are also provided. As outputs, the program produces sample allele counts at the specified

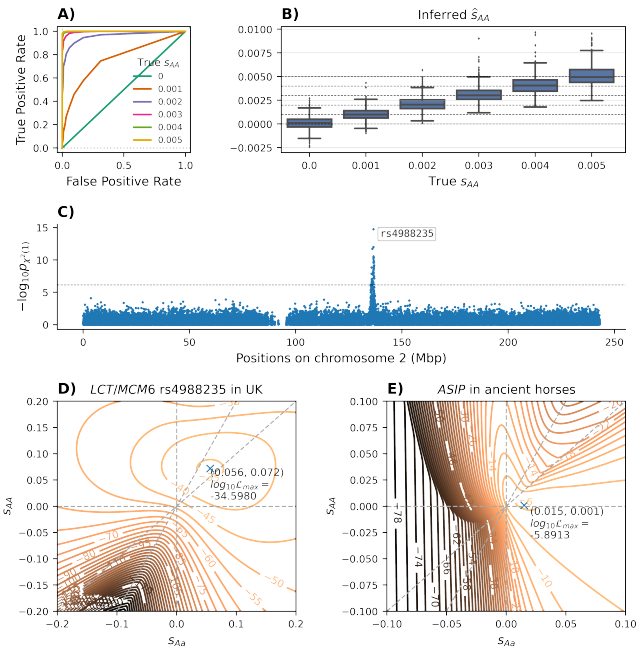

Figure 1: **A)** Receiver operating characteristic (ROC) curves exhibiting the power to distinguish replicates simulated under additive selection from neutral ones using likelihood-ratio tests. Curves based on replicates of 9 temporal samples of 20 diploid individuals, each 500 generations apart, and the diploid population size is $N_e = 10^4$. **B)** Distributions of MLEs $\hat{s}_{\mathrm{AA}}$ for the same set of simulated data in A) (details in Section S2.1 of *Supplementary Note*). **C)** Manhattan plot of p-values for filtered SNPs on Chromosome 2 in temporally-stratified ancient human samples from the United Kingdom. The dashed line indicates the negative logarithm of the Bonferroni-corrected threshold $p = 0.05$. Details are described in Section S4 in *Supplementary Notes*. **D**) and **E**) Log-likelihood surface across a two-dimensional grid of selection coefficients s_{Aa} (for heterozygotes; x-axis) and s_{AA} (for homozygotes; y-axis), where likelihoods are computed on 101×101 linear grids, respectively with **D)** SNP rs4988235 in the *LCT/MCM6* locus from the ancient human dataset used in C), and **E)** *ASIP* locus in ancient horses from Ludwig et al. (2009). Diagonal dashed gray lines mark full dominance ($s_{\mathrm{Aa}} = s_{\mathrm{AA}}$) and additivity ($s_{\mathrm{Aa}} = s_{\mathrm{AA}}/2$).

sampling times, and, optionally, their population allele frequency trajectories. Lastly, the program also allows for convenient visualization of the sample and population frequency trajectories for select replicates.

2.2 Python module `diplo_locus`

The module `diplo_locus.likelihood` contains the class `SelHmm`, which initializes the HMM for computation using specified population and selection parameters as described in Section 2.1. The function `SelHmm.computeLikelihood()` then computes per-locus log-likelihoods based on these parameters for given temporal samples at many loci. The module `diplo_locus.simulate` contains the function `simulateSamples()` to simulate independent replicates based on given population and selection parameters. It returns replicate samples with the corresponding underlying trajectories. Both, analysis and simulation can accommodate piece-wise constant temporal changes in selective pressure.

3 Performance Evaluation

We tested the performance of `diplo-locus` on data simulated using either `diplo-locus` (Fig. 1A, Fig. 1B, and Section S2 in *Supplementary Notes*), or `SLiM4` (Haller and Messer, 2023) (see Section S3 in *Supplementary Notes*), as well as well-characterized empirical data (Fig. 1C, Fig. 1D, Fig. 1E, and Section S4 in *Supplementary Notes*). In simulations, the likelihood-ratio of the estimated MLEs divided by neutrality ($s = 0$) computed using `diplo-locus` can distinguish replicates simulated under selection from neutral replicates (Fig. 1A). Furthermore, the MLEs of s_{AA} capture the true value used for the simulations (Fig. 1B). The MLEs of diploid selection coefficients for the two empirical datasets are in line with previous studies: Heterozygote advantage for *ASIP* in horses (Fig. 1E), analyzed, for example, by Steinrücken et al. (2014), and positive selection in the UK population at the SNP `rs4988235` (Fig. 1D, shown to regulate the lactase *LCT* gene expression (Troelsen et al., 2003), consistent with previous findings (Mathieson and Terhorst, 2022). Moreover, the tests for selection around the *LCT* locus are most significant among all variants on chromosome 2 (Fig. 1C, Fig. S13 and S14). Here, we extracted human samples in the UK dated to be more recent than 4,500 years before present from the Allen ancient DNA Resource Database (Mallick et al., 2023, v54.1), see Section S4.2 in *Supplementary Notes* for details.

Our tool `diplo-locus` reuses calculations efficiently: When computing likelihoods on a grid, the transition matrices for the underlying HMM can be shared across loci, resulting in substantial computational speed-up. As a benchmark, using a single core of an Intel Ice Lake processor (2.9GHz) and 32GB RAM, it takes ∼5.5 hours to compute log-likelihoods for the 520 samples from the UK genotyped at 69,903 SNPs (after filtering) on Chromosome 2 across a 51-point grid of s_{AA} values (Fig. 1C and Section S4.2 in *Supplementary Notes*). However, because all the computed log-likelihoods are stored in memory before writing output or maximizing, the program might consume more memory than the user's system capacity in cases of a large number of loci or dense parameter grids. Thus, to analyze a larger number of loci, it might be necessary to split the analysis into several parallel batches.

4 Conclusion

We present a toolkit developed in Python for analyzing temporal genetic samples under the Wright-Fisher model, with a focus on inferring general diploid selection. Our software can compute log-likelihoods for time series data, perform likelihood-based inference, and simulate replicates of such data. We assessed the efficiency and accuracy of the inference using our software and showcased the versatility of the CLI `DiploLocus` and the python module `diplo_locus`. We believe this package will be a welcome addition to the toolkit of the population genetics community.

Acknowledgements

We thank Adam Fine, Jeremy Berg, John Novembre, Maanansa Raghavan, and Constanza de la Fuente Castro for helpful discussions and feedback on this project.

Funding

This work is supported by the University of Chicago, a Leakey Foundation Research Grant, and by the National Institute of General Medical Sciences (NIGMS) of the National Institutes of Health under award number R01GM146051.

References

J. P. Bollback, T. L. York, and R. Nielsen. Estimation of $2N_e s$ from temporal allele frequency data. *Genetics*, 179(1):497–502, 2008.

B. C. Haller and P. W. Messer. SLiM 4: multispecies eco-evolutionary modeling. *The American Naturalist*, 201(5):E000–E000, 2023.

C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, et al. Array programming with NumPy. *Nature*, 585:357–362, 2020.

Z. He, X. Dai, M. Beaumont, and F. Yu. Estimation of natural selection and allele age from time series allele frequency data using a novel likelihood-based approach. *Genetics*, 216(2):463–480, 2020.

A. Ludwig, M. Pruvost, M. Reissmann, N. Benecke, G. A. Brockmann, P. Castaños, M. Cieslak, S. Lippold, L. Llorente, A.-S. Malaspinas, et al. Coat color variation at the beginning of horse domestication. *Science*, 324(5926):485–485, 2009.

A.-S. Malaspinas, O. Malaspinas, S. N. Evans, and M. Slatkin. Estimating allele age and selection coefficient from time-serial data. *Genetics*, 192(2):599–607, 2012.

S. Mallick, A. Micco, M. Mah, H. Ringbauer, I. Lazaridis, I. Olalde, N. Patterson, and D. Reich. The allen ancient dna resource (aadr): A curated compendium of ancient human genomes. *bioRxiv: https://doi.org/10.1101/2023.04.06.535797*, 2023.

I. Mathieson and G. McVean. Estimating selection coefficients in spatially structured populations from time series data of allele frequencies. *Genetics*, 193(3):973–984, 2013.

I. Mathieson and J. Terhorst. Direct detection of natural selection in Bronze Age Britain. *Genome Research*, 32(11-12):2057–2067, 2022.

S. G. Self and K.-Y. Liang. Asymptotic properties of maximum likelihood estimators and likelihood ratio tests under nonstandard conditions. *Journal of the American Statistical Association*, 82(398):605–610, 1987.

M. Steinrücken, A. Bhaskar, and Y. S. Song. A novel spectral method for inferring general diploid selection from time series genetic data. *The Annals of Applied Statistics*, 8(4):2203–2222, 2014.

M. Steinrücken, E. M. Jewett, and Y. S. Song. Spectraltdf: transition densities of diffusion processes with time-varying selection parameters, mutation rates and effective population sizes. *Bioinformatics*, 32(5): 795–797, 2016.

J. T. Troelsen, J. Olsen, J. Møller, and H. Sjöström. An Upstream Polymorphism Associated with Lactase Persistence has Increased Enhancer Activity. *Gastroenterology*, 125(6):1686–1694, 2003.

L. S. Whitehouse and D. R. Schrider. Timesweeper: accurately identifying selective sweeps using population genomic time series. *Genetics*, 224(3):iyad084, 05 2023. ISSN 1943-2631. doi: 10.1093/genetics/iyad084. URL `https://doi.org/10.1093/genetics/iyad084`.