`Moments LD` user manual

Corresponding to version 0.0.1

Aaron Ragsdale
Contact: aaron.ragsdale@mail.mcgill.ca

April 4, 2019

Contents

1 Introduction to `mold` 4
1.1 Getting help and helping us . . . 4

2 LD statistics 4
2.1 Hill-Robertson statistics . . . 5
2.2 Multi-population LD statistics . . . 5

3 Getting started 5
3.1 Installation . . . 5
3.1.1 Dependencies . . . 5
3.1.2 Downloading and compiling `moments` and `moments.LD` . . . 5
3.2 Suggested workflow . . . 6

4 Parsing and importing data 6
4.1 `LDstats` objects . . . 6
4.2 Estimating two-locus statistics from data . . . 6
4.3 Parsing data from a genotype matrix . . . 6
4.4 Parsing a VCF file . . . 6
4.4.1 Using a recombination map . . . 6
4.4.2 Restricting based on features . . . 6
4.5 Creating bootstrap datasets . . . 6

5 Specifying a model 8
5.1 Implementation . . . 8
5.2 The `Demography` builder . . . 8
5.3 Units . . . 8

6 Simulation and fitting the model 10
6.1 Running the model . . . 10
6.2 Comparing model to data . . . 10
6.2.1 Likelihoods . . . 10
6.2.2 Fitting . . . 10
6.3 Uncertainty analysis . . . 10

7 Plotting **10**
7.1 Visualizing LD curves 10
7.2 Residuals 10

8 The full two-locus frequency spectrum **10**
8.1 `moments.TwoLocus` 10
8.1.1 Specifying models 10
8.1.2 Parameters 10
8.1.3 Selection 10

9 Frequently asked questions **10**

10 Acknowledgements **10**

Example code

1 **Parsing:** Example of parsing data generated by msprime 7
2 **Bottleneck:** At time `TF + TB` in the past, an equilibrium population goes through a bottleneck of depth `nuB`, recovering to relative size `nuF`. 9

1 Introduction to `mold`

Welcome to `moments.LD`, a program for simulating linkage disequilibrium statistics. `moments.LD`, or `mold`, can compute a large set of informative LD statistics for many populations, and performs likelihood-based demographic inference using those statistics.

There are three primary features of `mold` to enable LD-based demographic inference: reading and parsing data, building demographic models, and inferring the parameters of those models by comparing model predictions to data. Typically, we use biallelic SNP data, along with a recombination map, to compute two-locus statistics over a range of genetic distances. We then use `mold` to compute expectations for those statistics under the demographic models we want to test, which can include multiple populations with variable migration, splits and mergers, and population size changes. Using a likelihood-based inference approach, we optimize those models to find the set of parameters that best fit the data.

I've tried to make parsing data and defining demographic models as painless as possible, though the complexity of the program does require some amount of script-writing and interaction. Luckily, `mold` is written in Python, a friendly and powerful programming language. If you are already familiar with ∂a∂i or *moments*, or Python in general, you are in a good position to dive right in to `mold`. If you have limited Python experience, this manual should provide the background and examples to get you up to speed and productive with `mold`.

1.1 Getting help and helping us

Undoubtedly, there will be bugs. If you find a bug in `mold`, or more generally if you find certain aspects of the program to be unintuitive or difficult to use, we would appreciate the feedback. Please submit a bug report at `https://bitbucket.org/simongravel/moments/issues`, and we will try to address the issue in a timely manner. Similarly, if you have suggestions for improved functionality or feature requests, those can be submitted in the issues as well or you can contact me directly.

As we do our own research, *moments* and `mold` are constantly improving. Our philosophy is to include any code we develop for our own projects that may useful to others. If you develop *Moments*-related code that you think might be useful to others, please let us know so we can include it with the main distribution.

2 LD statistics

Patterns of linkage disequilibrium (LD) are informative about evolutionary history, for example for inferring recent admixture events and population size changes or localizing regions of the genome that have experienced recent selective events. LD is commonly measured as the covariance (or correlation) of alleles co-occurring on a haplotype. The covariance (D) is

$$\begin{aligned} D = \mathrm{Cov}(A, B) &= f_{AB} - pq \\ &= f_{AB}f_{ab} - f_{Ab}f_{aB}, \end{aligned}$$

and the correlation (r) is

$$r = \frac{D}{\sqrt{p(1-p)q(1-q)}}.$$

We think of expectations of these quantities as though we average over many realizations of the same evolutionary process, but in reality we have only a single observation for any given pair of

SNPs. Therefore in practice we take the averages of LD statistics over many independent pairs of SNPs.

$\mathbb{E}[D]$ is zero genome wide, so LD is often measured by the variance of D ($\mathbb{E}[D^2)$ or the square correlation (r^2), where

$$r^2 = \frac{D^2}{p(1-p)q(1-q)}.$$

Because it is difficult to compute expectations for $\mathbb{E}[r^2]$ under even simple evolutionary scenarios, and because it is difficult to accurately estimate $\widehat{r^2}$ from data, we use $\mathbb{E}[D^2$ and related statistics to compare model predictions for LD to data.

2.1 Hill-Robertson statistics

Hill and Robertson (1968) introduced a recursion for $\mathbb{E}[D^2]$ that allows for variable recombination rate between loci and population size changes over time. To solve for $\mathbb{E}[D^2]$, this system requires additional LD statistics, which we call $Dz = D(1-2p)(1-2q)$ and $\pi_2 = p(1-p)q(1-q)$, where p and q are the allele frequencies at the left and right loci, respectively. This system also relies on heterozygosity (H), so from this system we can compute the vector of statistics

$$y = \begin{pmatrix} \mathbb{E}[D^2] \\ \mathbb{E}[Dz] \\ \mathbb{E}[\pi_2] \\ \mathbb{E}[H] \end{pmatrix}.$$

Instead of computing $\mathbb{E}[r^2]$, which is an expectation of ratios, Hill and Robertson (1968) and Ohta and Kimura (1971) studied the related statistic $\sigma_D^2 = \frac{\mathbb{E}[D^2]}{\mathbb{E}[\pi_2]}$. This statistic has the advantage that its expectation can be computed from the Hill-Robertson recursion, and we can accurately compute it from either phased or unphased data.

2.2 Multi-population LD statistics

In Ragsdale and Gravel (2018), we extend

3 Getting started

3.1 Installation

3.1.1 Dependencies

1. `numpy`, `scipy`, what else...
2. For parsing, we take advantage of `scikit-allel` ()
3. For demography building: `networkx`

3.1.2 Downloading and compiling `moments` and `moments.LD`

1. Can be cloned from (pulling to get updates)
2. Commands for installing

3.2 Suggested workflow

1. Python strengths: interactive
2. I often have two windows open: one is a script file and the other is IPython, which allows me to interactively test code, and then record it in script
3. importing `import moments.LD as mold`
4. Magic commands
5. Reloading modules that you've changed
6. Running script from command line

4 Parsing and importing data

1. `mold` represents two-locus statistics using `mold.LDstats` objects (describe what these are) APR: add rs to LDstats attributes, can store LD stats for multiple recombination distances
2. To create an LDstats object, we could just call `y = mold.LDstats([[0.001,0.0005,0.002],[0.05]])` .
3. Typically, we either compute the LDstats object from a demographic model (below), or we build the object from data.
4. `mold` can create an LDstats object given a vcf file (and optionally a recombination map, mask files, and population files), for either phased or unphased data
5. Or given a genotype array

4.1 `LDstats` objects

1. Attributes
2. Marginalization, swapping

4.2 Estimating two-locus statistics from data

(Ragsdale and Gravel, 2019)

4.3 Parsing data from a genotype matrix

1. All loci, not caring about recombination distance (say, all unlinked loci)
2. with the recombination distance between snps known

4.4 Parsing a VCF file

4.4.1 Using a recombination map

4.4.2 Restricting based on features

4.5 Creating bootstrap datasets

```
this is a script dot py
```

Listing 1: **Parsing:** Example of parsing data generated by msprime

5 Specifying a model

General overview of demographic models

5.1 Implementation

Manual input of demographic model (gets difficult with more than two populations)

5.2 The `Demography` builder

5.3 Units

1. mutation rate
2. recombination rate
3. unit of time
4. N_e and N_{ref}

```
def bottleneck(params, ns):
    nuB, nuF, T = params
    nu_func = lambda t: [nuB * numpy.exp(numpy.log(nuF/nuB)
                                 * t / T)]

    sts = moments.LinearSystem_1D.steady_state_1D(ns[0])
    fs = moments.Spectrum(sts)
    fs.integrate(nu_func, T)

    return fs
```

Listing 2: **Bottleneck:** At time `TF + TB` in the past, an equilibrium population goes through a bottleneck of depth `nuB`, recovering to relative size `nuF`.

1. With old approach: bottleneck with growth, IM model
2. With demography: Gutenkunst model, Archaic model (w/ denisovans, neand, papuan, EA, EU, Afr)

6 Simulation and fitting the model

6.1 Running the model

6.2 Comparing model to data

6.2.1 Likelihoods

6.2.2 Fitting

1. Parameter bounds
2. Fixed parameters
3. Other options (there are a lot more now)
4. optimizer choice

6.3 Uncertainty analysis

7 Plotting

7.1 Visualizing LD curves

7.2 Residuals

Based on expectations, observations, and covariances

8 The full two-locus frequency spectrum

8.1 `moments.TwoLocus`

8.1.1 Specifying models

8.1.2 Parameters

8.1.3 Selection

9 Frequently asked questions

APR: No one has really asked me questions yet, but here are my own quandaries and answers:

1.
2. How do I cite `mold`?
3. What if I'm having issues running this program?
   Bug: issues
   Bigger issues or difficulties: email

10 Acknowledgements

1. Ryan Gutenkunst
2.

References

Hill, W. G. and Robertson, A. (1968). Linkage disequilibrium in finite populations. *Theoretical and Applied Genetics*, 38(6):226–231.

Ohta, T. and Kimura, M. (1971). Linkage disequilibrium between two segregating nucleotide sites under the steady flux of mutations in a finite population. *Genetics*, 68(4):571–580.

Ragsdale, A. P. and Gravel, S. (2018). Models of archaic admixture and recent history from two-locus statistics. *BioRxiv*, doi:10.1101/489401.

Ragsdale, A. P. and Gravel, S. (2019). Unbiased estimation of linkage disequilibrium from unphased data. *BioRxiv*, doi:10.1101/557488.