est-sfs. Estimation of the unfolded site frequency spectrum

Peter D. Keightley

Institute of Evolutionary Biology, University of Edinburgh, Charlotte Auerbach Rd, Edinburgh EH9 3FL, UK

Documentation for version 2.04:
Written 6th February 2018
Updated 3rd July 2019.

1. Introduction

est-sfs is a stand-alone implementation of a method to infer the unfolded site frequency spectrum (the uSFS) and ancestral state probabilities by maximum likelihood (ML). The uSFS is a vector of counts of sites with *x* derived allele copies in a sample of *n* gene copies from a population. est-sfs infers the uSFS and ancestral state probabilities using information from up to three outgroups. Three models of nucleotide substitution are implemented. For details of the method, see Keightley and Jackson (2018).

2. Citing est-sfs

If you use the program and publish a paper, then please make an appropriate citation.

Basis of the approach:

Keightley, P. D., Campos, J. L., Booker T. R. and Charlesworth, B. (2016). Inferring the frequency spectrum of derived variants to quantify adaptive molecular evolution in protein-coding genes of *Drosophila melanogaster*. *Genetics* **203**: 975-984.

Generalized version of the method:

Keightley, P. D and Jackson B. C. (2018). Inferring the probability of the derived versus the ancestral allelic state at a polymorphic site. *Genetics* **209**: 897-906.

3. Installation

The program requires the GNU Scientific Library (gsl). See http://www.gnu.org/software/gsl/ to download gsl and to find out how to install it on your system.

Having downloaded the source code along with its Makefile, a single executable object file is created by the gcc compiler using the make command:

```
$ make
```

The software is written in C to be compiled on a Linux system, and should also run on other Unix-based systems such as Mac OS X. It is unknown whether the software runs on Windows systems: you could try a Unix emulator.

4. Running the program

The program is run if by invoking the object file name (assuming the current directory is in your path) with input and output files specified in the following order on the command line:

```
$ est-sfs config-file.txt data-file.txt seed-file.txt
        output-file-sfs.txt output-file-pvalues.txt
```

5. Input and output files

5.1. Configuration file (config-file.txt)

This file (examples provided) consists of three lines of text in the following order:

n_outgroup [1, 2, or 3]
model [0, 1 or 2]
nrandom [0 or positive integer]

model

0 = Jukes-Cantor model
1 = Kimura 2-parameter model
2 = Rate-6 model (see Keightley and Jackson 2018 for details)

nrandom

If nrandom = 0, the ML search algorithm starts with pre-set parameter starting values. If nrandom >= 1, the algorithm returns the highest log likelihood found in nrandom runs using starting values for the parameters randomly sampled from wide ranges. The MLs of the individual runs are reported. In the case of the Rate-6 model, it is recommended that at least 10 random starting value runs are carried out to ensure convergence.

5.2. Data file (data-file.txt)

An example data file, TEST-DATA.TXT, for three outgroups is provided. The phylogenetic tree topology must conform to that shown in Fig. 1 (with one, two or three outgroups included in the data file).

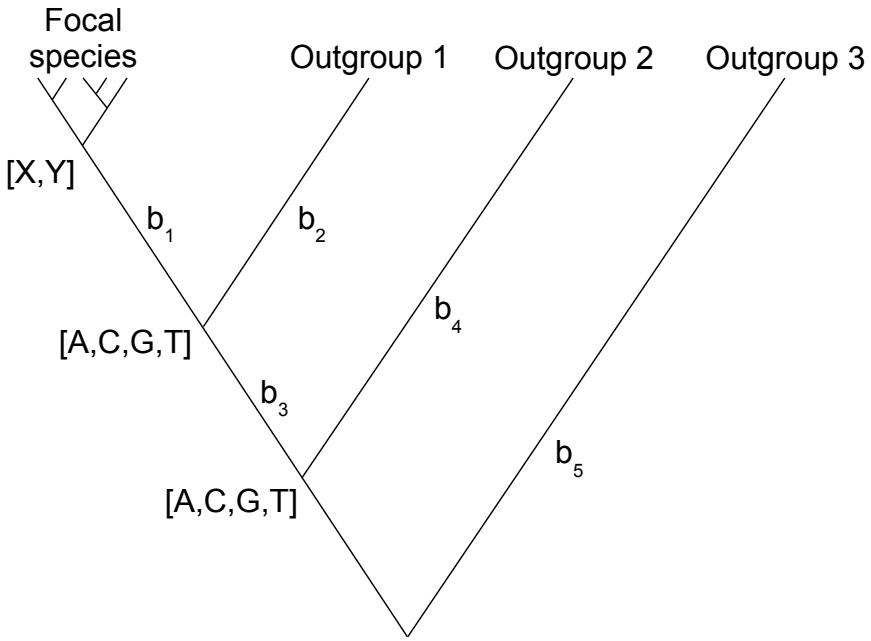

Figure 1.

If there are three outgroups, there are 4 space-separated columns. The first column is for the focal species, and the next three columns are for the outgroups. Each column is a comma-separated list of the counts of the four bases in the order A, C, G, T. For example, the first line in the example data file is:

```
20,0,0,0        0,0,0,1 0,0,0,1 0,0,0,1
```

At this site, all $n = 20$ copies sampled in the focal species are A. In the three outgroups, a single copy has been sampled, and in each case it is T. All sites must have the same number of copies sampled in the focal species and up to one copy sampled in each outgroup. If there are missing data in any outgroup, the counts for that outgroup are encoded `0,0,0,0` . Data from polymorphic and non-polymorphic sites are analysed together.

5.3. Seed file (`seed-file.txt`)

This text file (example provided) contains a single positive integer. It is overwritten by a new random number seed at the end of the run.

5.4. uSFS output file (`output-file-sfs.txt`)

This output file consists of the comma-separated estimated uSFS vector containing $n + 1$ elements.

5.5. Ancestral state probabilities output file (`output-file-pvalues.txt`)

Among other things, this file contains the estimated ancestral state probabilities for each site. Also output is the maximum log likelihood of the model, which may be compared to the log likelihood for alternative nucleotide substitution models.

The file begins with several lines started by "0" containing various self-explanatory outputs from the program. There then follow numbered lines, containing fields as follows:

1. Line number. This corresponds to the line number of the data file.
2. Configuration index. Not of interest to the user.
3. The probability of the major allele being ancestral.
4-7. For two outgroups: the probabilities of the first internal node (Fig. 1) having states A, C, G or T.
4-19. For three outgroups: the probabilities of the first and second internal nodes (Fig. 1) having states [A,A], [A,C], [A,G], [A,T], [C,A], [C,C], [C,G], [C,T], etc.