User manual for *MuteBaSS*

Xiaoheng Cheng and Michael DeGiorgio

April, 2018

Contents

1 Introduction

MuteBaSS is a set of Python scripts to perform scans with summary statitics HKA, NCD, NCD_{opt}, NCD_{sub}, and NCD_{mid}, for detecting footprints of long-term balancing selection affecting one or more species (Cheng and DeGiorgio). Operation of this package requires a UNIX environment with Python 2.7 and above.

Please cite it as:

X. Cheng and M. DeGiorgio. Detection of shared balancing selection in the absence of trans-species polymorphism. *bioRxiv* doi:XXXX

If you experience any issues, please contact Xiaoheng Cheng at xh.cheng@psu.edu for further help.

2 Operation

2.1 Installation

We distribute *MuteBaSS* in compressed (tar.gz) format. In addition to the *MuteBaSS* scripts, we also included the user manual and example data. The scripts included are designed to perform on a UNIX system. To unpack `MuteBaSS` from the command line, go to the directory where it is stored, and enter

```
tar -xzvf MuteBaSS.tar.gz
cd MuteBaSS/
```

The first command will decompress the file and release the content into folder `MuteBaSS/` in the current directory. The second command will lead the user to the `MuteBaSS/` directory, which contains the manual, `test` directory, and three Python scripts: `MuteBaSS.py`, `HKAtrans.py`, and `transNCDs.py`. When using `MuteBaSS.py`, please make sure to keep these three python scripts in the same directory.

2.2 Performing scans

To run `MuteBaSS.py`, format the dash commands and their arguments in a single line as

```
python MuteBaSS.py -i <input file> -c <p,x> [--check] [--tree <tree>]
   [--fixSize] -w <window size> -s <step size> -o <output file>
   [--NCD] [--tf <tf>] [--NCDopt] [--NCDsub] [--NCDmid]
   [--HKA] [--config <config file>] [--getConfig]
```

In this command line, the paths and names of the input and output files (*i.e.*, arguments `<input file>` and `<output file>`) follow the dash commands `-i` or `--input`, and `-o` or `--output`, respectively. The command -c or `--indices`, is used to inform of the input file layout. Following the command, user must provide the column number for physical positions (denoted as `p`), and that of the first column showing allele counts (denoted as `x`) in the input file, in the format of `<p,x>`, without space (find more details in Section 3). The `--check` command can be used to check whether the input file has correct format (see more in Section 2.3).

To run the scan, the size of windows (*i.e.*, `<window size>` argument) for which summary statistics will be calculated should be provided following `-w` command. If the user chooses to adopt scanning windows of fixed length (using `--fixSize`), then the number following `-w` should be the length in bases, *e.g.*, `-w 1500` is for a 1.5 kilobase [kb] window. Otherwise, the window size should be the number of informative sites flanking the test site on either side, *e.g.*, use `-w 30` for a window containing 30 informative sites on either side of the test site, totalling 61 sites within the window. When scanning with HKA, NCD, NCD_{opt}, or NCD_{sub}, users need to use the command `-s` to set the step size (*i.e.*, `<step size>`), which should be the length in bases between the center of neighboring windows if `--fixSize` is used. Otherwise, it should instead be the distance in number of informative sites between neighboring test sites. For scans with NCD_{mid}, the windows

will center on every polymorphic site, and therefore do not need the knowledge of step size.

To choose the set of summary statistics to be computed, users can use a combination of dash commands `--HKA`, `--NCD`, `--NCDopt`, `--NCDsub`, and `--NCDmid`. Because the NCD statistic requires a pre-determined target frequency (argument `<tf>`), when choosing to perform scans with NCD, users should assign the target minor allele frequency using command `--tf`. If not, then the NCD statistic will be calculated based on the default target frequency of 0.5. Further, because NCD_{mid} adopts distinct scanning windows from all other statistics, users are not advised to perform NCD_{mid} scans in combination with others. In cases where it is included in the combination, separate output files will be generated for NCD_{mid} and others, and the one for NCD_{mid} will have the suffix `_snpCT.txt`, or `_fixSizeCT.txt` when users choose to fix the window size in terms of physical distance.

When choosing to perform a HKA scan on input data, a corresponding configuration file (argument `<config file>`, following the command `--config`) is needed to inform `MuteBaSS.py` of the fractions (conditional on informative sites) of within-species polymorphisms and between-species substitutions in neutral (*e.g.*, whole genome) scenarios. Format requirements for configuration files can be found in Section 3. When needed, `MuteBaSS.py` can help generate the configuration file given the neutral input file. To this end, `--getConfig` can be used as followes, where the path and name to the configuration file must be provided with `--config` command.

```
python MuteBaSS.py -i <input file> -c <p,x> --getConfig --config <config file>
```

2.3 Checking input file format

The command argument `--check` is optional. When the user wishes to check if the input file has the correct format before scanning, provide input file (`-i`) and the column indices (`-c`), and include `--check` in the command.

```
python MuteBaSS.py -i <input file> -c <p,x> --check
```

When the input file comprises genomic data from more than three species, there exists more than one unrooted bifurcating topology relating to these species. Because the summary statistics included in `MuteBaSS.py` only consider substitutions among all species fitting the species tree that relates them, it is important to ensure that input data are compatible with their relationships. In this case, users should include the `<tree>` argument, which describes the tree topology, following the `--tree` command.

```
python MuteBaSS.py -i <input file> -c <p,x> --check --tree <tree>
```

To construct a `<tree>` argument, the K species concerned are represented by integers $1, 2 \ldots, K$, in the same order that they are presented in the input file. With these numbers, the tree will then be presented via Newick notation (see examples in Figure 1). Rooting the tree is not mandatory. Note that this argument must be presented between quotation marks. That is, the left and right trees in Figure 1 would be represented as `"(((1,2)3),4)"` and `"(((1,2),3),(4,5))"`, respectively, in place of the `<tree>` argument.

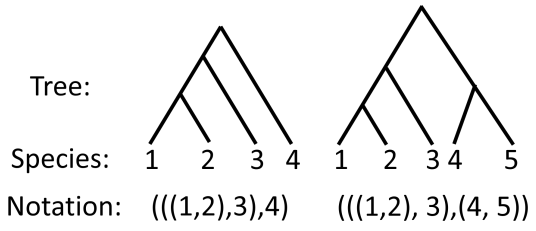

Figure 1: Illustration of Newick notations of phylogenic trees.

To reduce the computing load, while performing scans with the assigned summary statistics, `MuteBaSS.py` assumes all input files have correct formats, and will not check whether the observed counts across all species match the species tree. Therefore, we highly recommend that users check the format of input files with the `--check` command before beginning their analyses.

2.4 Help page

The user can use the `-h` or `--help` commands to display instructions for each dash command. Either of the two commands should print out the following output in the command window. This help page would also appear if no arguments are provided.

```
Usage:  python MuteBaSS.py -i <input file> -c <p,x> [--check] [--tree <tree>]
[--fixSize] -w <window size> -s <step size> [--HKA][--config <config file>]
[--getConfig] [--NCD][--tf <tf>] [--NCDopt] [--NCDsub] [--NCDmid] -o <output file>

Options:
-h, --help            show this help message and exit
-i INFILE, --input=INFILE
                      Path and name of your input file.
-o OUTFILE, --output=OUTFILE
                      Path and name of your output file.
-n INDEX, --indices=INDEX
                      Index numbers for the columns of locus positions p
                      and allele counts of the first population x,
                      respectively.  Format of this argument should be "p,x",
                      without space.
--fixSize             Option to fix the size of scanning windows.  When true,
                      provide the length of weindow in bp with "-w" command.
-w R, --window=R      Number of sites flanking the test locus on either
                      side.  When choose to fix window size (--fixSize),
-s STEP, --step=STEP  For windows with fixed number of sites, each step is
                      the number of sites between each neighboring test
                      sites.  For windows with fixed length, each step is the
                      length, in bp, between neighboring test sites.
--HKA                 Option to perform HKA scan.  User must provide (with "
                      --config") configuration file, which records the
                      fractions of substitutions and polymorphisms among all
                      informative sites.
--config=CONFIGFILE   Path and name of your configuration file for HKA scan.
                      File should be tab-delimited, with the first k columns
                      showing sample sizes for each species, then two
                      columns for the corresponding fractions of within-
                      species polymorphisms and substitutions, respectively.
--getConfig           Option to generate configuration file from
                      concatenated whole-genome (neutral) input file.
--NCD                 Option to perform NCD scan.  Default target frequency
                      (tf) is 0.5.  User can customize the value with "--tf".
--tf=TF               Target frequency for NCD scan.  Default tf is 0.5.
                      tf must be no greater than 0.5.
--NCDopt              Option to perform NCDopt scan.
--NCDsub              Option to perform NCDsub scan.
--NCDmid              Option to perform NCDmid scan.
--check               Option to check the format of input file.
--tree=TREE           Tree topology if given more than 3 species.
```

3 Input format

`MuteBaSS.py` requires allele count data from all species to be examined on within-species polymorphisms and cross-species substitutions. To present such data, input files must be tab-delimited plain text files, with at least columns recording physical positions, ancestral allele count, and total allele count of each site.

Input and output files are recommended to be presented in absolute paths. If their absolute paths are unavailable, then users can copy the scripts `MuteBaSS.py`, `HKAtrans.py`, and `transNCDs.py`, to the same folder where the data are located, and follow the instructions in Sections 2 or 5.

If the input file has been previously edited in an operating system environment other than UNIX, then we advise users to use the following command to ensure this file is readable in a UNIX environment:

```
dos2unix <file>
```

3.1 Input file format

The input file must at least include, in addition to the physical position of each informative site, the number of ancestral alleles (denoted as `x`) and the total number of alleles sampled (denoted as `n`). To be an informative site, a site should either be polymorphic in only one of the species examined, or be monomorphically different across all species, with the pattern agreeing with the species tree. Sites that fit neither of these two types should be discarded. All sites should be bi-allelic. All input files should include one-line headers, otherwise the first line will be automatically excluded from analyses. To examine K species, the input file should at least contain the following columns

```
position  x1  n1  ...  xk  nk
```

where `position` is for physical positions, and `xj` and `nj` denote the ancestral and total allele counts, respectively, at this position in species j, $j = 1, 2, \ldots, K$. Site positions should be presented in ascending order.

`MuteBaSS.py` accepts input files with additional columns to the aforementioned essential ones, and the argument `-c <p,x>` ensures that the software can find the information needed. For example, users should input `-c 2,5` for an input file with the following header.

```
Chr  pos  anc  drv  x1  n1  ...  xk  nk
```

Importantly, all additional information should be presented before the columns with allele counts. In other words, `nk` should mark the last column for the input with K species. To check the format of input files, use `--check` command after providing the column indices (see Section 2.3).

3.2 Configuration file for HKA scan

When choosing to perform scans with the HKA statistic, users must provide a corresponding configuration file. This file records the proportions (conditional on informative sites) of within-species polymorphisms and cross-species substitutions for each set of sample sizes. Configuration files should not have headers, should be tab-delimited, and each line should present the needed information in the following order:

```
<n1>  <n2>  ...  <nk>  <%poly>  <%sub>
```

where `<nj>` represents the total number of alleles observed in species j, $j = 1, 2 \ldots, K$, `<%poly>` stands for the fraction of polymorphic sites among all informative sites with this set of sample sizes, and `<%sub>` for that of substitutions. In general, for a file for K species, it should include $K + 2$ columns, with the first K columns showing the sample sizes, in the same order that these species are presented in the input file, and two subsequent columns for the fractions of polymorphisms and substitutions, respectively. Users can check `test/forkSp-HKA_config.txt` included in the package as references for configuration files for $K = 1$ to 4 species.

4 Output format

`MuteBaSS.py` writes output to the path and file name provided in the `-o <output file>` argument as a tab-delimited plain text table. The first column indicates the location of the corresponding window with which the statistics are computed. For sliding windows with fixed physical length, it is shown as the position of the center of each window (*i.e.*, `midPos`). For windows containing fixed number of informative sites, the position is that of the central site (*i.e.*, `sitePos`). For scans with NCD_{mid}, positions would be those of the polymorphic sites that it centers on at each step (*i.e.*, `snpPos`). Dependent on the set of statistics users choose to compute, the output file would include one column for each statistic, with each row being the values computed for the corresponding window. Additionally, for scans with fixed physical length window, output will include the number of sites covered by the window (*i.e.*, `numSites`). Moreover, if users choose to scan with NCD variants integrated with optimization, then `MuteBaSS.py` will output the optimal frequency at each step (*i.e.*, `optF`). For NCD_{mid}, the software will also output the number of polymorphic sites covered by the window (*i.e.*, `numSNPs`), as well as the frequency of the center polymorphic site at each step (*i.e.*, `fc`).

All output files will have headers. Assuming the user opted to compute all statistics in a single scan, when choosing a fixed physical length for the window size, the header of the output for HKA, NCD, NCD_{opt}, and NCD_{sub} will be

```
midPos   HKA   NCD   NCDopt   NCDsub   numSites   optF
```

Note that when both NCD_{opt} and NCD_{sub} are computed, their optimal target minor allele frequencies will both be presented in the `optF` column, and be separated by commas. The output file for NCD_{mid}, whose name ends with `_fixSizeCT.txt`, will have header

```
snpPos   NCDmid   numSites   numSNPs   fc
```

Alternatively, when each window contains a fixed number of informative sites, the output file for HKA, NCD, NCD_{opt}, and NCD_{sub} will have header

```
sitePos   HKA   NCD   NCDopt   NCDsub   optF
```

Similarly, the output file for NCD_{mid}, with the suffix `_snpCT.txt`, will begin with

```
snpPos   NCDmid   numSNPs   fc
```

5 Examples

To further illustrate the usage of *MuteBaSS*, we included example input, configuration, and output files in the subfolder `test/` for K species, where $K = 1$ to 4. All example inputs are parsed from a SLiM (Messer, 2013) simulation where five species evolve along the tree displayed in Figure A of Cheng and DeGiorgio. The input files with one, two, three, and four species, respectively, begin with

```
==> test/testin_1sp.txt <==
 rep  position  x  n
 42   35.0      0  50
 42   46.0      0  50
 42   64.0      0  50
 ...
==> test/testin_2sp.txt <==
 position  x1  n1  x2  n2
 35.0      0   50  44  50
 74.0      50  50  0   50
 99.0      50  50  0   50
 ...
```

```
==> test/testin_3sp.txt <==
 physPos   genPos   x1   n1   x2   n2   x3   n3
 6         0.0012   50   50   50   50   0    50
 19        0.0038   0    50   50   50   50   50
 24        0.0048   50   50   25   50   50   50
 ...
==> test/testin_4sp.txt <==
 rep   physPos   genPos   x1   n1   x2   n2   x3   n3   x4   n4
 42    6         0.0012   50   50   50   50   0    50   50   50
 42    19        0.0038   0    50   50   50   50   50   50   50
 42    24        0.0048   50   50   25   50   50   50   50   50
 ...
```

The column indice argument (*i.e.*, `-c <p,x>`) for these files should be `-c 2,3` for `test/testin_1sp.txt`, `-c 1,2` for `test/testin_2sp.txt`, `-c 1,3` for `test/testin_3sp.txt`, and `-c 2,4` for `test/testin_4sp.txt`.

In addition to the input files, the `test/` folder also includes configuration files for each type of input, named `forkSp-HKA_config.txt`, where `k` is the number of species presented in the input.

5.1 Performing scans with HKA, NCD, NCD_{opt}, and NCD_{sub}

To perform scans with HKA, NCD, NCD_{opt}, and NCD_{sub}, include the corresponding command for each statistic accordingly, and provide the software with window size parameters. For example, to detect balancing selection affecting four species using all four statistics with a window size of 2,000 bases and step size of 500 bases, with 0.4 as the target frequency for NCD, the command line should be

```
python MuteBaSS.py -i test/testin_4sp.txt -c 2,4 --fixSize -w 2000 -s 500
      --HKA --config test/for4Sp-HKA_config.txt --NCD --tf 0.4 --NCDopt
      --NCDsub -o testout_4sp_HKA-NCD4-NCDopt-NCDsub_2kb.txt
```

Note that users do not need to use all four statistics for analyses. For example, to scan through two-species input with only NCD variants, use the default target frequency of 0.5 for NCD, adopt sliding windows with 20 informative sites on either side of each test site, and take steps of every 5 informative sites, the following command line should be used.

```
python MuteBaSS.py -i test/testin_2sp.txt -c 1,2 -w 20 -s 5 --NCD --NCDopt
      --NCDsub -o testout_2sp_NCD-NCDopt-NCDsub_20sites.txt
```

Similarly , the following command can be used to scan through the three-species input computing only HKA and NCD, with fixed window size of 2,000 bases and step size of 500 bases, and adopting a target frequency of 0.3.

```
python MuteBaSS.py -i test/testin_3sp.txt -c 1,3 --fixSize -w 2000 -s 500
      --HKA --config test/for3Sp-HKA_config.txt --NCD --tf 0.3
      -o testout_3sp_HKA-NCD3_2kb.txt
```

Likewise, to apply NCD and NCD_{opt} on the one-species input with a window size of 30 sites on either side of test sites and a step size of 5 sites, the command line below can be used.

```
python MuteBaSS.py -i test/testin_1sp.txt -c 2,3 -w 30 -s 5 --NCD --NCDopt
   -o testout_1sp_NCD-NCDopt_30sites.txt
```

The output files from these examples are provided in the `test/` folder, and the user's outputs following the above commands should be identical with the sample files.

5.2 Performing scans with NCD_{mid}

Similar to including HKA and other NCD variants in the scan, the `--NCDmid` command can be used to choose to compute NCD_{mid}. However, because it requires that each window be centered on a polymorphic site, the sliding windows it adopts are different from all other statistics. We therefore recommend users not to mix the command for NCD_{mid} with those of other statistics. Hence, to scan through the two-species input file with a window size of 20 informative sites on either side of the central polymorphic site, the command line should be

```
python MuteBaSS.py -i test/testin_2sp.txt -c 1,2 -w 20 --NCDmid
      -o testout_2sp_NCDmid_20sites.txt
```

To perform NCD_{mid} scan on the four-species input with a window size of 2,500 bases, the following command line can be used.

```
python MuteBaSS.py -i test/testin_4sp.txt -c 2,4 --fixSize -w 2500
      --NCDmid -o testout_4sp_NCDmid_2.5kb.txt
```

Note that the actual names of the output files for these two examples are `testout_2sp_NCDmid_20sites_snpCT.txt` and `testout_4sp_NCDmid_2.5kb_fixSizeCT.txt`, respectivley. The two sample output files are also included in `test/` folder, and should be identical to the user's outputs.

References

X. Cheng and M. DeGiorgio. Detection of shared balancing selection in the absence of trans-species polymorphism. *Submitted*.

P. W. Messer. SLiM: simulating evolution with selection and linkage. *Genetics*, 194(4):1037–1039, 2013.