SeleDiff v1.0 Manual

Xin Huang

Friday, March 16, 2018

Contents

`SeleDiff` is implemented with a probabilistic method for estimating and testing selection (coefficient) differences between populations[1].

If you use `SeleDiff`, please cite

```
Huang X, Jin L, He Y. 2018. SeleDiff: A fast and scalable tool for
estimating and testing selection differences between populations. *In submission*.
```

1 The Model

Consider a bi-allelic locus in the population i, let $p_i(t)$ and $q_i(t)$ denote the derived and ancestral allele frequencies at time t, respectively. We can define the absolute fitness of the derived and ancestral alleles as w_D and w_A, respectively. We then define the relative fitness as

$$e^s = \frac{w_D}{w_A}.$$

Here, s is the (allele) selection coefficient. Based on our previous study[1], the selection (coefficient) difference between populations i and j is

$$\begin{aligned} d_{ij} &= s_i - s_j \\ &= \frac{1}{t}\left[\ln \frac{p_i(t)/q_i(t)}{p_j(t)/q_j(t)} + \Omega\right], \\ &= \frac{1}{t}\left[\ln OR + \Omega\right] \end{aligned}$$

where OR stands for odds ratio; Ω has a mean of zero, and reflects the uncertainty of allele frequencies caused by factors other than selection; and t is the divergence time from populations i and j to their most recent common ancestor. The expectation and variance of d_{ij} is

$$\begin{aligned} E(d_{ij}) &= \frac{1}{t}\ln OR \\ \mathrm{var}(d_{ij}) &= \frac{1}{t^2}\left[\mathrm{var}\left(\ln OR\right) + \mathrm{var}\left(\Omega\right)\right] \end{aligned}.$$

d_{ij} is approximately normal, because $\ln OR$ and Ω approximately follows normal distributions. Its 95% confidence interval is $E\left(d_{ij}\right) \pm 1.96\sqrt{\mathrm{var}\left(d_{ij}\right)}$. We proposed a statistic δ for testing selection differences in a locus:

$$\delta = \frac{\left[E\left(d_{ij}\right)\right]^2}{\mathrm{var}\left(d_{ij}\right)}.$$

Under the null hypothesis, $E(d_{ij}) = 0$, thus, δ follows a central χ^2-distribution with one degree of freedom. Under the alternative hypothesis, δ follows a non-central χ^2-distribution with non-centrality parameter $E(d_{ij})$ with one degree of freedom. Because

$$\begin{aligned} \delta &= \frac{\left[E\left(d_{ij}\right)\right]^2}{\mathrm{var}\left(d_{ij}\right)} \\ &= \frac{\ln^2 OR}{\mathrm{var}\left(\ln OR\right) + \mathrm{var}\left(\Omega\right)} \sim \chi_1^2 \end{aligned},$$

and the median of χ^2_1-distribution approximately equals to 0.455. Therefore, given a dataset with n loci, we assume most loci are neutral in both populations i and j, i.e. $s_i = 0 = s_j, E(d_{ij}) = 0$. Then we can estimate $\mathrm{var}(\Omega)$ as

$$\mathrm{var}(\Omega) = \mathrm{median}\left\{\frac{\ln^2 OR_k}{0.455} - \mathrm{var}(\ln OR_k), 1 \le k \le n\right\}.$$

where

$$\mathrm{var}(\ln OR) \approx \frac{1}{N_i\hat{p}_i(t)} + \frac{1}{N_i\hat{q}_i(t)} + \frac{1}{N_j\hat{p}_j(t)} + \frac{1}{N_j\hat{q}_j(t)}.$$

Here, N_i and N_j are the sample sizes of populations i and j. We add 0.5 to allele counts less than 5 for continuity correction.

2 Usages

2.1 Installation

To install `SeleDiff`, you should install Java SE Development Kit 8 or OpenJDK8 first.

After the installation, you can check Java version in the command line (command starts by ">" prompt).

```
> java -version
java version "1.8.0_25"
Java(TM) SE Runtime Environment (build 1.8.0_25-b17)
Java HotSpot(TM) 64-Bit Server VM (build 25.25-b02, mixed mode)
```

2.1.1 Linux/Mac

In Linux/Mac, you can open the terminal and clone `SeleDiff` using `git`:

```
> git clone https://github.com/xin-huang/SeleDiff
```

Then you can enter the `SeleDiff` directory and use `gradlew` to install `SeleDiff`:

```
> cd ./SeleDiff
> ./gradlew build
> ./gradlew install
```

The runnable `SeleDiff` is in `./build/install/SeleDiff/bin/`. You can add this directory into your system environment variable `PATH` by:

```
> export PATH="/path/to/SeleDiff/build/install/SeleDiff/bin/":$PATH
```

You can get help information by typing:

```
> SeleDiff
```

and you will get the following:

```
Usage: SeleDiff [command] [command options]
  Commands:
    compute-var      Sub-command for estimating variances of Omega
      Usage: compute-var [options]
        Options:
          --geno
```

```
        The EIGENSTRAT GENO file stores allele counts: 0, zero copy of the
        reference allele; 1, one copy of the reference allele and one copy
        of the alternative allele; 2, two copies of the reference allele;
        9, missing values.
    * --ind
        The EIGENSTRAT IND file stores information of individuals and
        populations.
    * --output
        The output file.
      --snp
        The EIGENSTRAT SNP file stores information of variants.
      --vcf
        The VCF file stores SNP information and genotype data.

compute-diff      Sub-command for estimating selection differences of loci
  Usage: compute-diff [options]
    Options:
      --geno
        The EIGENSTRAT GENO file stores allele counts: 0, zero copy of the
        reference allele; 1, one copy of the reference allele and one copy
        of the alternative allele; 2, two copies of the reference allele;
        9, missing values.
    * --ind
        The EIGENSTRAT IND file stores information of individuals and
        populations.
    * --output
        The output file.
      --snp
        The EIGENSTRAT SNP file stores information of variants.
    * --time
        The file stores divergence times between populations. A divergence
        time file is space delimited without header, where the first
        column is the population ID of the first population, the second
        column is the population ID of the second population, the third
        column is the divergence time of this population pair. This file
        is needed when estimating selection differences.
    * --var
        The file stores variances of Omega, which is space delimited
        without header the first column is the first population ID the
        second column is the second population ID the third column is the
        variance of drift of this population pair. This file is needed
        when estimating selection differences.
      --vcf
        The VCF file stores SNP information and genotype data.
```

* indicates required options

You can use `gradlew` to remove `SeleDiff`:

```
> ./gradlew clean
```

2.1.2 Windows

In Windows, you can download the latest release. Please make sure your environment variable `JAVA_HOME` correctly point to you JDK directory. After download and uncompression, you can open `cmd` and enter the directory of `SeleDiff` in `cmd`. Please use `gradlew.bat` to build and install `SeleDiff`:

```
> cd /path/to/SeleDiff
> gradlew.bat build
> gradlew.bat install
```

And run `SeleDiff.bat` in `./build/install/SeleDiff/bin/`:

```
> cd /build/install/SeleDiff/bin/
> SeleDiff.bat
```

You can use `gradlew.bat` to remove `SeleDiff`:

```
> cd /path/to/SeleDiff
> gradlew.bat clean
```

2.2 Commands

`SeleDiff` contains two sub-commands:

- `compute-var` for estimating variances of Ω^1, which is required for the `compute-diff` command;
- `compute-diff` for estimating selection differences among loci.

2.3 Input Files

`SeleDiff` assumes bi-allelic genetic data and will not perform any checks on this assumption. All input files can be compressed by `gzip`.

2.3.1 EIGENSTRAT

`SeleDiff` accepts EIGENSTRAT format of genetic data as inputs. EIGENSOFT provides several functions to convert other formats to EIGENSTRAT format.

For EIGENSTRAT format, there are 3 files: **SNP** file, **IND** file and **GENO** file. Consider an example dataset containing 3 unrelated individuals (Ind1, Ind2 & Ind3) from 3 populations (Pop1, Pop2 & Pop3) that were typed on 3 SNPs (SNP1, SNP2 & SNP3):

```
        SNP1 SNP2 SNP3
Ind1    T/T  A/T  T/T
Ind2    C/G  C/G  C/G
Ind3    C/C  A/A  ?/?
```

The **SNP** file describes the information of each SNP. The SNP file corresponding to the example dataset is:

```
SNP1 1 0.1 100 A T
SNP2 1 0.2 101 C G
SNP3 1 0.3 103 C A
```

Each row corresponds to a SNP. The 6 columns are:

1. SNP ID
2. Chromosome number
3. SNP genetic position

4. SNP physical position
5. Reference allele
6. Alternative allele

The **IND** file describes the information of each individual. The IND file corresponding to the example dataset is:

```
Ind1 M pop1
Ind2 F pop2
Ind3 U pop3
```

Each row corresponds to an individual. The 3 columns are:

1. Individual ID
2. Sex: M for male, F for female and U for unknown
3. Population ID

The **GENO** file contains genetic data. The GENO file corresponding to the example dataset is:

```
010
111
209
```

Each row corresponds to a SNP, and each column corresponds to an individual. The characters, 0, 1, 2, 9, correspond to an individual's genotype:

- 0 means zero copies of reference allele.
- 1 means one copy of reference allele.
- 2 means two copies of reference allele.
- 9 means missing data.

2.3.2 VCF

`SeleDiff` also accepts VCF format of genetic data as inputs, and assumes genotypes of each individual are encoded with 0 and 1. Because VCF format contains no population information of each individual, users should provide an additional file following EIGENSTRAT IND format.

2.3.3 Var File

The Var file is the output file from the first sub-command `compute-var`, which stores variances of pairwise Ω. `SeleDiff` does not divide Ω with generation times as He et al. (2015)[1] in order to reduce floating-point rounding errors. When using sub-command `compute-diff` to estimate selection differences, `SeleDiff` uses `--var` option to accept a a *SPACE* delimited file without header that specifies variances of Ω between two populations.

```
YRI CEU 1.547660
YRI CHS 1.639591
CEU CHS 0.989241
```

The first two columns are the population IDs, and the third column is the variances of Ω of the two populations.

2.3.4 Divergence Time File

When using sub-command `compute-diff` to estimate selection differences, `SeleDiff` uses `--time` option to accept a *SPACE* delimited file without header that specifies divergence times between two populations.

```
YRI CEU 5000
YRI CHS 5000
CEU CHS 3000
```

The first two columns are the population IDs, and the third column is the divergence times of the two populations.

2.4 Output File

The output file from `SeleDiff` is *TAB* delimited. The first row is a header that describes the meaning of each column.

Column	Column Name	Description
1	SNP ID	The name of a SNP
2	Ref	The reference allele
3	Alt	The alternative allele
4	Population1	The first population ID
5	Population2	The second population ID
6	Selection difference	The selection difference between the first and second populations
7	Std	The standard deviation of the selection difference
8	Lower bound of 95% CI	Lower bound of 95% confidence interval of the selection difference
9	Upper bound of 95% CI	Upper bound of 95% confidence interval of the selection difference
10	Delta	The δ statistic for selection difference
11	p-value	The p-value of the δ statistic

3 An Example

Here is an example to show how `SeleDiff` estimates and tests selection differences between populations. 4 populations (YRI, CEU, CHB, CHD) from HapMap3 (release3) were extracted. CHB and CHD were merged into one population called CHS. PLINK 1.7 were used to remove correlated individuals and SNPs with minor allele frequencies less than 0.05 and strong linkage disequilibrium. These genome-wide data are stored in `./examples/data/example.geno` and used for estimating variances of Ω.

Two alternative alleles (rs1800407 and rs12913832) associated with blue eyes were identified in genes *HERC2* and *OCA2*.[2] These candidate data are stored in `./examples/data/example.candidates.geno` and used for estimating selection differences of these SNPs between populations.

The counts of alleles in our example data were summarized in below.

SNP ID	Population	Reference Allele Count	Alternative Allele Count
rs1800407	YRI	290	0
rs1800407	CEU	207	17
rs1800407	CHS	486	4
rs12913832	YRI	294	0
rs12913832	CEU	47	177
rs12913832	CHS	491	1

We assume the divergence time of YRI-CEU and YRI-CHS are both 5000 generations, while the divergence time of CEU-CHS is 3000 generations. This information is stored in `./examples/data/example.time`.

First, we estimate variances of Ω using sub-command `compute-var`.

```
> SeleDiff compute-var --geno ./examples/data/example.geno \
                       --ind ./examples/data/example.ind \
                       --snp ./examples/data/example.snp \
                       --output ./examples/results/example.geno.var
```

To estimate selection differences, we use the sub-command `compute-diff`.

```
> SeleDiff compute-diff --geno ./examples/data/example.candidates.geno \
                        --ind ./examples/data/example.candidates.ind \
                        --snp ./examples/data/example.candidates.snp \
                        --var ./examples/results/example.geno.var \
                        --time ./examples/data/example.time \
                        --output ./examples/results/example.candidates.geno.results
```

The result is stored in `./examples/results/example.candidates.geno.results`. The main result is in below.

SNP ID	Population1	Population2	Selection difference	Std	delta	p-value
rs1800407	YRI	CEU	-0.000773	0.000380	4.129	0.042154
rs1800407	YRI	CHS	-0.000336	0.000393	0.731	0.392559
rs1800407	CEU	CHS	0.000728	0.000377	3.730	0.053443
rs12913832	YRI	CEU	-0.001541	0.000378	16.583	0.000047
rs12913832	YRI	CHS	-0.000117	0.000415	0.080	0.777297
rs12913832	CEU	CHS	0.002372	0.000433	30.062	0.000000

From the result, we can see the selection coefficient of rs12913832 in CEU is significantly larger than that in YRI or CHS, which indicates rs12913832 is under directional selection in CEU. While the selection coefficient of rs1800407 in CEU is marginal significantly larger than that in YRI or CHS.

4 Dependencies

- Java 1.8
- Apache Commons Math 3.6
- JCommander 1.72
- t-digest 3.1

5 References

1. He Y, Wang M, Huang X, Li R, Xu H, Xu S, Jin L. 2015. A probabilistic method for testing and estimating selection differences between populations. *Genome Res*, **25**: 1903-1909.
2. Sturm RA, Duffy DL, Zhao ZZ, Leite FP, Stark MS, Hayward NK, Martin NG, Montgomery GW. A single SNP in an evolutionary conserved region within intron 86 of the *HERC2* gene determines human blue-brown eye color. *Am J Hum Genet*, **82**: 424-431.