

## **polyHap2 Documentation**

### **Environment Set Up**

- Java Runtime Environment is required in order to run polyHap2. The latest version of JRE can be downloaded from Sun website. polyHap2 is designed to work with version 1.6.0\_01 JRE or later.

### **To Execute Program**

- Download the polyHap2.zip from website. This zip file contains:
  - (1) main program---**polyHap2.jar**
  - (2) one log file, **log.txt**.
  - (3) one batch file, **run.bat**.
  - (4) some examples of input files, output files and command lines.
  - (5) Library files (\* .jar), java library files used by our main program
- Put these files in the same directory, including two input files---genotype data and sample files. Then run the executable batch file, 'run.bat', which contains two command lines:

(1) `java -cp polyHap2.jar dataFormat.CompressPolyHap2 --paramFile log.txt`

This command line will convert your input files (genotype and sample files) to the format which is needed for the polyHap2. This produces a **.zip** file containing SNP information and a build file.

(2) `java -cp polyHap2.jar lc1.dp.appl.DickFormat --paramFile log.txt`

This performs the main program of polyHap2 Note, before executing the main program, make sure the zip file and the build file are included in the same directory.

Or, you could create a batch file on your own, which includes these two command lines. The memory can be modified by the argument, `-Xmx1000m`.






Then the command line will look something like this:

`Java -Xmx1000m -cp polyHap2.jar lc1.dp.appl.DickFormat --paramFile log.txt`

### **Input File Format**

- **log file** specifies the parameters used in the program, such as input file names, chromosome name, phasing types (non-internal or internal phasing) ,the SNP region included for analysis, the number of chromosome (ploidy)...etc. Please see the detail in the file, 'log.txt'.
- **Genotype data file**---each row contains information about SNP id and SNP

position and genotypes for each of individuals. Please note that this file should not have a header line. Each column is separated by a space or tab. At moment the code only deals with biallelic markers and a deletion. The first two columns are SNP id and SNP position, respectively. Then, the following columns are genotypes (one for each individual). For example, a row with 5 individuals from the dataset might look like:

SNP1	1000	GTG	G	TTG	TG	TG
						
		Ind 1	Ind 2	Ind 3	Ind 4	Ind 5

For each marker, alleles can be coded as any two different letters, apart from 'N' which is the code for missing allele. If the number of allele of a missing genotype, it should be coded as 'NA'. An example file is shown in 'genotypeInfo.txt'.

- **Sample file** consist of one column for the list of all sample id. The order of this list should be the same as in the genotype file. An example file is shown in 'sampleID.txt'.

## Output File

- **phased\_states.txt\_1**: contains the phased ancestral haplotypes (phased states). The first two rows are the number of individuals and SNPs respectively, following by the phased ancestral haplotypes for each of individuals. Each number represents a cluster. One row presents one haplotype. One column is for one marker.
- **Phased2.txt\_1**: contains the phased genotypes. The first two rows are the number of individuals and SNPs respectively, following by the phased genotypes for each of individuals. Alleles are shown in 'A', 'B' or '\_', where 'A' presents the first allele shown in the dataset for each SNP. One row presents one haplotype. One column is for one marker.
- **Phased1.txt\_1**: contains certainty rates for each estimate. The first two columns are SNP id and position respectively. The following columns give the certainty rate for each of individuals. Each individual has three columns which are the certainty rate for phased states, phased genotypes, predicted genotypes (for missing genotype), respectively.