Users Manual for PoolHap v.0.1

28 July 2010

1. Introduction

PoolHap is a tool to infer the haplotype frequencies from pooled sequencing. It is built as a Java .jar package, which can run in all platforms (Windows, Mac OS X, & Linux) as long as JVM is installed. Please refer our paper "PoolHap: inferring haplotype frequencies from pooled samples by next generation sequencing" for the descriptions of the design and algorithms.

2. Installation

Please download PoolHap package, PoolHap.tar.gz from our website. By typing

$ tar –xvzf PoolHap.tar.gz

in the terminal of Mac or Linux environment or using WinRAR in Windows, one can decompress the package and see the following files in the folder PoolHap:

poolhap.jar, which is the executable;
sample_hap.txt, which is a sample of haplotype file.
sample_genemodel.gff, which is a sample of gene model file for RNA-Seq analysis.

The regression of PoolHap was done by another public available java library written by Michael T. Flanagan. So the users have to download that library as well. After downloading poolhap.jar, please download the package, flanagan.jar, from http://www.ee.ucl.ac.uk/~mflanaga/java/Regression.html and put it in a folder named poolhap_lib in the same folder where you put poolhap.jar. Please use the exact name "poolhap_lib" and don't change the name of flanagan.jar because the classpath of poolhap.jar has been set to poolhap_lib already.

Now you are ready to use PoolHap. By opening a terminal and typing

$ java –Xmx2g –jar poolhap.jar, the help information will be prompt. If you see:

```
$ java –Xmx2g -jar poolhap.jar
PoolHap: Infer haplotype frequencies from pooled sequencing
Version: 0.1; contact: quan.long@gmi.oeaw.ac.at
Usage: java -jar poolhap.jar <function> [options]
Functions:
        select   Select most informative SNPs for infer
        infer    Infer haplotype frequencies
        rnaseq   Infer isoform frequencies for RNA-Seq data
```

Then you have successfully installed PoolHap.

There are three sub-functions: **select**, **infer**, and **rnaseq**. "**select**" is the function to help users to select most informative SNPs when there are plenty of candidates, e.g., in case of whole genome sequencing. "**infer**" is the main function of PoolHap to get haplotype frequencies. "**rnaseq**" is the function specific for RNA-Seq data.

3. Usage

3.1. select

select is an optional function to help users to select most informative SNPs. Users can choose to use it or not.

File format: the input file (raw haplotypes with all SNPs) and output file (new haplotypes composed of selected SNPs) of **select** are in the same format. The file should contain 1+*h* lines, in which the first line is a header and the each other line describes a known haplotype. The first line is composed of SNP information separated by spaces. The format of each SNP information is in the form of Chr_name:loction. The rest lines are alleles separated by spaces. Please refer sample_hap.txt as an example.

When you type java –Xmx2g -jar poolhap.jar select, you will see

```
$ java –Xmx2g -jar poolhap.jar select
Select most informative SNPs
Usage: java -jar poolhap.jar select -in <input> -out <output> -n <#snps> [options]
Options:
        -diff <smallest_diff_threshold> [0.45]
        -round <max_round> [10000]
        -try <number_try> [20]
```

As one can see, there are maximal 6 parameters to be set for this function, in which three of them are mandatory: input file, output file, and number of SNPs to be selected. The other three are the parameters for the selection: -diff specifies the preferable smallest difference between any pair of haplotypes, -round specifies the times of iteration from the same initial guess, -try specifies the number of initial guesses.

Besides the above procedure of selecting informative SNPs, we also suggest that users select SNPs to avoid mapping errors, SNP calling errors, as well as structural rearrangements. The current mappers for NGS (e.g., BWA[1]) usually gives SNP quality and coverage of the SNPs. We suggest the users to select SNPs with high SNP qualities and mediate coverage. But we do not provide this function as part of PoolHap pipeline due to many existing mappers and SNP calling methodologies and file formats.

3.2 **infer**

This is the main function for inferring haplotype frequencies. When you type java –Xmx2g -jar poolhap.jar infer, you will see

$ java –Xmx2g -jar poolhap.jar infer
Infer haplotype frequencies from pileup data
Usage: java -jar poolhap.jar infer -knownhap <known haplotypes> -pileup <pileup file>

There are two mandatory parameters: one is –knownhap, the other is –pileup. -knownhap specifies known haplotypes, in the format of haplotype file described in the section **3.1 select**. –pileup specifies the pileup file assembled from pooled sequencing, in the format specified by SAM specification (http://samtools.sourceforge.net/SAM1.pdf).

If you are not familiar with reads mapping and SAM specifications, here are some basic introductions: using any mapper supporting SAM format (e.g., BWA, SOAP, Bowtie, and etc), one can map the reads to the reference genome to get the .sam or .bam file. After that, by SAMtools, one can generate consensus genome in the format of .pileup file.

Please notice that, in the two input files, the chromosome names and coordinates must follow the same system!!

After running the program, there will be a file named xxx.poolhap, which contains the results.

3.3 **rnaseq**

This is a sub-function for inferring relative abundance of different isoforms of same genes in RNA-Seq data.

Please note that the data this sub-function is working on is the isoforms generated by different gene models. In case the user wanted to get relative abundance of paternal and maternal transcripts by looking at heterozygotes SNPs in the mRNA, s/he will need to use the standard PoolHap, treating paternal/maternal transcripts as known haplotypes.

By typing java –Xmx2g -jar poolhap.jar rnaseq, you will see:

$ java –Xmx2g -jar poolhap.jar rnaseq
Infer haplotype frequencies from RNA-Seq data
Usage: java -jar poolhap.jar rnaseq -genemodel <Gene Model file> -pileup <pileup file> -ref <Reference Genome> -rd_len <read_length> -out <output file> [options]
Options:
    -max_iso <max_iso_num> [10]

There are five mandatory parameters: -genemodel specifies the .gff file telling PoolHap the known gene models. We include a typical .gff gene model file (sample_genemodel.gff) in the package. –pileup specifies the pooled sequences in the format of pileup file. This file can be generated by popular RNA-Seq mapping tool like Bowtie/TopHat plus SAMtools. –ref specifies the reference genome. –rd_len tells PoolHap the length of the reads. In case users have multiple libraries, one can just specify an average read length. PoolHap will use the read length for processing the

first and last exons only, so the precision of this parameter is not very crucial. –out specifies the output file.

An optional parameter is the maximal number of isoforms of genes.

Please notice that, like in the case of **infer**, in all three input files, the chromosome names and coordinates must follow the same system!!

4. Support & Feedback.

Please contact Quan Long (quan.long@gmi.oeaw.ac.at) for any other questions. Any feedback, e.g., file format suggestions, bugs report, and etc., will be grateful.

1 Li, H. & Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25**, 1754-1760, doi:btp324 [pii] 10.1093/bioinformatics/btp324 (2009).