Instruction to run LD estimation and Persistence of Phase calculations from phased files

Yvonne M. Badke
Department of Animal Science
Michigan State University
East Lansing, Mi, USA
email: badkeyvo@msu.edu

Juan P. Steibel
Departments of Animal Science, Fisheries and Wildlife
Michigan State University
East Lansing, Mi, USA

September 7, 2011
Version 1.0

1 Introduction

The scripts and haplotyes supplemented with this manual will allow the user to reproduce the results as presented in the accompanying paper [2]. The scripts will estimate r^2 as a measure of Linkage Disequilibrium (LD) calculating average r^2 for all pairwise combinations of SNP, average r^2 for adjacent SNP and correlation of phase for markers within a certain distance interval. The scripts can be transferred to any set of haplotypes given in the format described below.

The scripts consists of five separate .R files:

- input_information.R

- LDbycorrelation.R
- ld_average.R
- ld_functions.R
- PersistenceofPhase.R

If you are using these scripts or any part of it please cite [2] in all resulting publications.

2 Input data and formats

2.1 Directory structure

The scripts require a specific directory structure (Fig. 1), as indicated below:

1. Create a new directory (TopDirectory) and name it at your discretion.
2. Within the TopDirectory create one subdirectory per population.
3. Copy haplotype files from each population (see section 2.2 for format) into the corresponding subdirectories.
4. Within each subdirectory create a file called phased_filenames.txt containing the filenames of your haplotype files. This file has no header and should contain one filename per row.
5. Copy the five script files into TopDirectory.
6. Copy the map-file in the TopDirectory (see section 2.3 for format).

2.2 Haplotype Files

The scripts require haplotype data in BEAGLE [4] output format [3]. In brief, haplotypes are given as one file per chromosome (Fig. 2). The first column is an identifier 'I', the second column contains the SNP name and all following columns contain the transmitted and untransmitted haplotype per individual.

Haplotypes can be encoded using either A/T/C/G, A/B or any numerical code, with only biallelic markers being accepted by current code.

Figure 1: Directory Structure

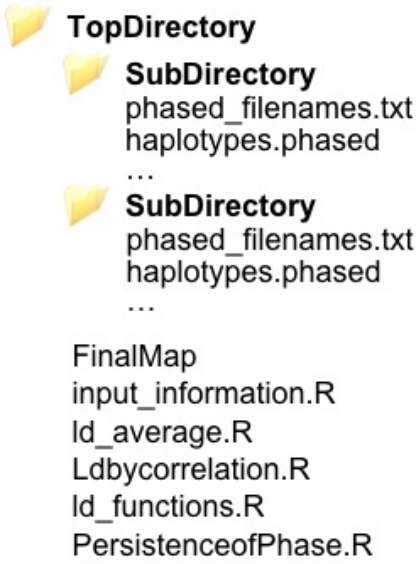

Figure 2: Input format of haplotypes

I	SNP	ID1_1	ID1_2	....	IDn_2
M	SNP1	A	G	....	C
M	SNP2	T	A	....	G
M	...	...	...	....	...
M	SNPm	C	G	....	A

2.3 Map-file format

The file containing the map (Fig. 3) should be in TopDirectory, with three columns, 1) for the SNP name, 2) the chromosome and 3) the position on the chromosome in basepairs. The map needs to include all the SNP used in the haplotype files. The map can contain additional information such as non-used SNP or additional columns, which will be ignored.

Figure 3: Format of the map file

SNP Identifier	Chromosome	position in base pairs
...	...	...

2.4 Input Parameterfile input_information.R

The file input_information.R allows the user to set up analysis options. In this section the structure of this file is described. The file is an R script containing two types of lines:

1. Lines starting with # are comments and can be ignored.
2. Lines with assignment options `variable_name<-value`. To specify new settings the user will have to replace the `value` of several variables. Do not alter the `variable_name`.

Enter the name of your map file as a value of the variable `map_path` and a vector of chromosome names as values of `chr` (Fig. 4).

Figure 4

```
# map file location: main directory which should be the current directory
# file structure: tab-delimited with 3 columns:
# 1. SNP name, 2. chromsome (integer), 3. map position in bp
map_path<-"FinalMAP.txt"

## Specify the name of the indidividual chromosomes
#- we have 18 autosomes and use the integers 1-18 as names
chr<-1:n_chr
```

Specify the correct population structure assigning the names of the population folders as values to `breed_fold`, with their respective population identifier in `breed_name` and one color per population for plotting purposes as values of `cols` (Fig. 5).

The script will automatically estimate average r^2 for distance intervals of 100kb from 0 to 10Mb, and consequently plot r^2 by distance.

To compute average r^2 for other distance intervals please enter these into the vector `target_dist` (Fig. 6). In `target_dist` consecutive values will be considered as intervals, given they are an appropriate distance interval: i.e. $c(0, 10000, 50000, 40000, 60000)$ will yield the following intervals:

Figure 5

```
## number of breeds in your design
# please enter the name of the folders containing the breed specific information
breed_fold<-c(4,5,6,7)
# please enter the name of the breeds as you wish it to be printed in the figures
breed_name<-c("Duroc", "Hampshire", "Landrace", "Yorkshire")
# please enter the name of the colors you would like to use to
# represent your breeds in the same order as the breeds
# make sure to assign 1 color per breed
cols<-c("red", "green", "blue", "black")
```

0, 10000	appropriate interval	0-10kb
10000, 50000	appropriate interval	10-50kb
50000, 40000	not appropriate	no output produced
40000, 60000	appropriate interval	40-60kb

Figure 6

```
# this code will automatically produce output with average r2
#for averages distances 0-10B in 100kb incremental windows
# in addition this code will give average r2 for a variety of specific target distances
# - you may alter the vector below to change these,
#   ld will be calculated for all intervals in the vector with
#   consequtive numbers establishing an interval
target_dist<-c(50000, 60000, 450000, 550000, 950000, 1050000, 4950000, 5050000)
```

To set the names of the .pdf files with the graphical output of average r^2 between adjacent markers per chromosome and the decay of average r^2 by distance specify the value of `plot_by_chr` and `plot_by_dist` (Fig. 7).

Figure 7

```
# filename for figure of average LD by chromosome
plot_by_chr<-("Average_LD_by_chr.pdf")
# filename of figure of average LD by distance
plot_by_dist<-("Average_LD_by_dist.pdf")
```

To estimate average r^2 between adjacent SNP for sparse SNP panels set `run_sparse<-TRUE` (Fig. 8) and select the density of markers within the panel. In this example the vector sparse is set to 2, 4 and 10 which will results into 3 sets with every second, every fourth and every tenth marker being included respectively.

Figure 8

```
## other possible statistics ##
# LD for adjacent markers in a sparse set of SNP #
# please indicate below if you wish to derive sparse set of SNP
# by consequtively deleting markers from the map #
# please enter 'TRUE' for yes and 'FALSE' for no
run_sparse<-TRUE
# if you wish to estimate sparse sets please enter the number of SNP to leave out in each
set #
# i.e. if you enter c(2, 10) then two sets of sparse markers will be derived,
# the first using every second marker in the map, the second using every 10th marker in the
map
sparsing<-c(2,4,10)
```

Assign 1 color per combination of populations to use in figures of persistence of phase as values for `cols2`. Equivalent to the vector `target_dist` you can design a vector `target` (Fig. 9) to estimate persistence of phase between populations at specific distance intervals. The correlation of phase for intervals of 100kb length for markers with pairwise distance between 0 and 10Mb will automatically be calculated and provided as output.

Figure 9

```
## Persistence of Phase
# please enter the name of the colors you would like to use to
# represent your breeds in the same order as the breeds
# make sure to assign 1 color per breedwise combination 2 out of n_breeds : 4!/(2!*2!)
cols2<-c("red", "green", "blue", "black", "purple3", "turquoise4")
# the same distance intervals as specified for r2 will be applied, 0 to 10Mb in 100kb
increments
# target distances - special distances to compare i.e. to results of previous publications,
# intervals will be derived consequetive from these distances: (10, 50, 100) will yield 0-
10, 10-50, 50-100
target<-c(0, 10000, 50000, 60000, 100000)
```

Once you have made all adjustments necessary please make sure your input_information.R file is saved in the TopDirectory.

3 Running the Script and Output

3.1 Estimation of r^2 from haplotype files

To estimate r^2 for all possible pair of SNP and safe the output for further analysis run LDbycorrelation.R from within each populations subdirectory using

```
R --no-save -q < ../LDbycorrelation.R
```

when running R in BATCH mode or

```
setwd("your_path/TopDirectory/subdirectory")
source("../LDbycorrelation.R")
```

when running R interactively, where `your_path` is the full path to TopDirectory. This script will create a file SNP.txt containing a list of all SNP with haplotypes within a population. In addition, the tab separated text file LD.txt with the SNP combinations in the first two columns, r in the third column, pairwise distance in the fourth column and the chromosome in the fifth column (Fig. 10) will be created.

Figure 10: Structure of LD.txt

SNP1	SNP2	r	distance in base pairs	chromosome
...	...	...	...	...

This file will then be converted into an R ff object [1], creating LD.ff and LD.ff.ffData.

3.2 Estimation of average r^2 by distance

To estimate r^2 statistics the user will need to run the script ld_average.R in TopDirectory using either

```
R --no-save -q < ld_average.R
```

when running R in BATCH mode or

```
setwd("your_path/TopDirectory/")
source("ld_average.R")
```

when running R interactively, where `your_path` is the full path to TopDirectory.
This script will produce the following output files

1. Average_LD_by_chr.pdf
   A figure of average r^2 between adjacent markers per chromosome per population in the sample, where the chromosomes are on the x-axis and r^2 is on the y-axis.

2. Average_LD_by_dist.pdf
   A figure of average r^2 between markers with distance intervals of 100kb ranging from 0 to 10Mb. Each population in the sample will be plotted, where distance is on the x-axis and r^2 is on the y-axis.

3. LD_table1.txt
   This tab separated text file (Fig. 11) has one column per population, and r^2 statistics in the rows: average r^2 for adjacent marker, the percentage of adjacent marker-pairs with $r^2 > 0.2$ and $r^2 > 0.3$, the total number of markers, the average distance between markers, and one row per chromosome with the average r^2 for adjacent marker pairs on that chromosome.

Figure 11: Structure of LD_table1.txt

Statistic	Pop_1	Pop_2	..	Pop_n
r^2 adjacent	..	..	..	..
$\%r^2 > 0.2$	..	..	..	..
$\%r^2 > 0.3$	..	..	..	..
# SNP	..	..	..	..
Average marker distance	..	..	..	..
r^2 adjacent CHR 1	..	..	..	..
...	..	..	..	..
r^2 adjacent CHR k	..	..	..	..

4. LD_average.txt
   This tab separated text file (Fig. 12) contains average r^2 for each selected interval (see vector `target_dist` in Fig. 6) and for intervals of 100kb between 0 and 10Mb, with one column per population.

Figure 12: Structure of LD_average.txt"

Distance	Pop_1	Pop_2	..	Pop_n
target_dist	..	..	..	..
distance interval 0 - 100kb	..	..	..	..
...	..	..	..	..
distance interval 9.9Mb - 10Mb	..	..	..	..

5. LD_sparse.txt
   This tab separated text file (Fig. 13) contains r^2 statistics for sparse marker sets. It will have one columns per population as specified in Fig. 6 and one column specifying how many markers where skipped to create this panel. Per sparse panel five rows will be provided composed of the same information as those given in LD_table1.txt

Figure 13: Structure of LD_sparse.txt

Statistic	Pop_1	Pop_2	..	Pop_n	sparsing
r^2 adjacent	..	..	..	..	..
$\%r^2 > 0.2$	..	..	..	..	..
$\%r^2 > 0.3$	..	..	..	..	..
# SNP	..	..	..	..	..
Average marker distance	..	..	..	..	..

3.3 PersistenceofPhase.R

To estimate correlation of phase the user will need to run the script PersistenceofPhase.R in TopDirectory using either

```
R --no-save -q < PersistenceofPhase.R
```

when running R in BATCH mode or

```
setwd("your_path/TopDirectory/")
source("PersistenceofPhase.R")
```

when running R interactively, where `your_path` is the full path to TopDirectory.
This script will produce the following output files:

1. PersistenceofPhase_by_distance.txt
   This is a tab separated text file (Fig. 14) with persistence of phase for intervals specified in `target` (Fig. 9) with 1 column per population-combination and one row per distance interval specified. In addition, persistence of phase for intervals of 100kb ranging from 0 to 10Mb will be printed.

Figure 14: Structure of PersistenceofPhase.txt

Distance	Pop_1	Pop_2	..	Pop_n
target	..	..	..	..
distance interval 0 - 100kb	..	..	..	..
...	..	..	..	..
distance interval 9.9Mb - 10Mb	..	..	..	..

2. PercentPhasePersistence.txt
   This tab separated text file is structurally equivalent to PersistenceofPhase_by_distance.txt (see Fig. 14) and contains the percentage of markers in opposite phase within each previously described interval.

3. Timesincepopulationdiverged.txt
   This tab separated text file contains the population combinations and their estimated time since breed divergence.

4. PersistenceofPhase.pdf
   A figure of correlation of phase between markers within distance intervals of 100kb ranging from 0 to 5Mb. Each population comparison in the sample will be plotted, where distance is on the x-axis and correlation of phase is on the y-axis.

3.4 Expected Computation Time

Using a 32 core node with 256GB of RAM (Michigan State High Performance Computing Center, `www.hpc.msu.edu`) resources the whole analysis requires 50min of computation time.

On a Macintosh environment running R 2.13.1 [5] on a Lion operating system with 4GB RAM the complete analysis requires 130min of computation time.

3.5 Screen output

The script LDbycorrelation.R will print the current status:

```
DONE WITH CHROMOSOME k
```

The script will attempt to remove LD.txt and SNP.txt before starting. Should none of these files be in the directory a warning message will be printed by R, that these files could not be removed. The `ffsave` [1] operation at the end of the code will print 5 warnings:

```
[1] "  adding: tmp/Rtmp6i0j43/ffdf59d5700f.ff (deflated 99%)"
[2] "  adding: tmp/Rtmp6i0j43/ffdf2a4ba472.ff (deflated 67%)"
[3] "  adding: tmp/Rtmp6i0j43/ffdf46a54000.ff (deflated 8%)"
[4] "  adding: tmp/Rtmp6i0j43/ffdf2a21577d.ff (deflated 100%)"
[5] "  adding: tmp/Rtmp6i0j43/ffdf309c2c52.ff (deflated 100%)"
```

that are normal and do not indicate a failure in the code.

The script ld_average.R will print

```
In population breed_name the matrix length is: n_rows
```

for each population. In addition, a number of warnings will be printed by the ff library [1] used in this script. The following warnings are normal and not an indication of errors:

```
In FUN(c("tmp/Rtmpw1AGoL/ffdf54779b62.ff",
"tmp/Rtmpw1AGoL/ffdf44568b85.ff",  ... :
  NOTE: did not overwrite file 'tmp/Rtmpw1AGoL/ffdf54779b62.ff'

In `[.ff`(p, i2) : opening ff /tmp/RtmpJUPhzA/ffdf783228cb.ff
```

The script PersistenceofPhase.R will also produce the above mentioned warnings due to the use of the ff library [1].

References

[1] Daniel Adler, Christian Glaeser, Oleg Nenadic, Jens Oehlschlaegel, and Walter Zucchini. *Package "ff"*. Universitaet Goettigen, Goettingen, Germany, 2.2-3 edition, September 2009.

[2] Y. M. Badke, R. O. Bates, C. W. Ernst, C. Schwab, and J. P. Steibel. Estimation of linkage disequilibrium in four us pig breeds. *submitted*, 2011.

[3] Browning. *BEAGLE 3.3*. University of Washington, Department of Medicine, Division of Medical Genetics, February 2011.

[4] Brian L Browning and Sharon R Browning. A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals. *Am J Hum Genet*, 84(2):210–223, Jan 2009.

[5] R Team. *R: A language and environment for statistical computing*, 2009.