FISHR Documentation

Matthew C Keller
Associate Professor
Psychology and Neuroscience dept.
University of Colorado, Boulder

Doug Bjelland
Postdoctoral Trainee

Institute for Behavioral Genetics
1480 30th Street
Boulder, Colorado, 80303

August 29, 2016

1 Introduction

FISHR performs pairwise IBD segment detection from phased SNP data. The first portion of FISHR relies on a modification (GERMLINE2) of the GERMLINE program developed by Gusev et al. to find potential candidate segments. GERMLINE2 works by first cutting the chromosome into non-overlapping segments. Then, within each segment, an exact match is discovered between a single haplotype of one individual and a single haplotype of another individual. After the initial discovery is made, GERMLINE2 then extends the segment in both directions, allowing for certain situations of SNP errors and phase errors. Then, the FISHR step uses additional information to further refine the candidate segments detected by GERMLINE2. FISHR first combines any two segments that are separated by a user defined gap parameter. FISHR then finds all implied errors (IE) within the candi-

date segments. These are haplotype mismatches (potential phase errors or SNP errors) that are potentially inconsistent with IBD status. Once the IEs are discovered, a moving average of IEs is calculated for every SNP around a user defined `-window`. This moving average is used to determine the end positions of the called IBD segments; segments are terminated when the moving average is above user specified value of `emp_ma_threshold`. The last step for FISHR is to calculate the total proportion of IEs over the entire final called segment. If this value is greater than the user defined `-emp_pie_value`, the remaining segment is dropped altogether.

GERMLINE2 Input Files

`-pedfile` [file path] This is a standard plink *.ped file with one exception, it MUST be phased. Phased ped files look like normal ped files (with genotypes at each SNP separated by whitespace or tab), but in phased ped file the alleles to the left of the whitespace (or tab) are predicted to be on the same haplotype and similarly for those to the right of the whitespace (or tab). We have provided utility programs that can take a chromosome that was phased using Shapeit2 or BEAGLE and output a phased *.ped file. The paternal ID, maternal ID, sex, and disease status variables do not affect the FISHR output.
The columns correspond to: Family ID (FID), Individual ID (IID), Paternal ID, Maternal ID, Sex, Disease Status, allele1.SNP1 allele2.SNP1 allele1.SNP2 allele2.SNP2 allele1.SNP3 allele2.SNP3...
Here, allele1.SNP1, allele1.SNP2, and allele1.SNP3 are all predicted to be in phase (rest on the same haplotype), and similarly for allele2.SNP1, allele2.SNP2, and allele2.SNP3.

`-mapfile` [file path] This is a standard plink *.map file. The genetic map positions (cM) must be correspond as closely to the dataset being used to provide as accurate of estimates as possible.
The columns correspond to: Chromosome number, SNP name, Centimorgan distance, Basepair position, Major allele, Minor allele.

GERMLINE2 Parameters

-err_hom [integer] The maximum number of mismatching homozygous markers for a slice to still be considered part of a match.

-err_het [integer] The maximum number of mismatching heterozygous markers for a slice to still be considered part of a match

-bits [integer] This is the length of the initial potential segment, in SNPs, discovered by GERMLINE2 before any extension occurs. The chromosome is broken up into non-overlapping segments with a length in SNPs of the bits parameter. The initial segment discovery then finds exact matches between individuals of these non-overlapping segments. Smaller values (< 30) will result in more potential segments being discovered, and in turn, a much longer runtime. Longer values (> 120) will run much more quickly, but at the cost of missing potential segments

-min_m [numeric] The minimum length, in centimorgans, the segments discovered by GERMLINE2 need to be. So that FISHR can merge shorter segments that were incorrectly split by GERMLINE2, we recommend that this minimum cM length is less than the final desired cM length output by FISHR. For example, if you want to look at IBD segments ¿ 3 cM, then set this parameter to be, say, 1.5, and then set the minimum in FISHR to be 3 cM.

-homoz A flag. This is option has no parameters and is either included or not. If it is included, GERMLINE2 will attempt to detect runs of homozygosity within an individual in addition to detecting IBD segments between individuals.

-w_extend - A flag. If used alone (not in conjunction with h_extend), this extends the segment until the first opposite homozygote occurs in each direction. If used in conjunction with h_extend (as we recommend!), this option extends the segment until the number of phased mismatches exceeds some number, as defined by -err_het.

-h_extend - A flag. The option -h_extend should always be used in conjunction with -w_extend (so either use -w_extend alone, which uses opposite homozygotes to determine segment endpoints, which tend to be longer, or

use -h_extend and -w_extend together, in which case phase information is also used to determine segment endpoints). FISHR has been optimized to run with using -h_extend and -w_extend, but it's fine to use -w_extend alone if the user wishes to have more a higher false positive and lower false negative rate of initial candidate segments. FISHR will work fine for both, but tends to be more efficient and produce slightly more accurate results when both are used together.

-bin_out - A flag telling GERMLINE2 to write output in binary form. The output files will not be human readable therefore, but this saves a lot of space. This is necessary for being read by FISHR.

-reduced - A flag telling GERMLINE2 to reduce the columns of the output; necessary for being read by FISHR.

FISHR Input Files

-bmatch/-bmid/-bsid [file paths] - The binary files output from GERMLINE2 above (from the -bin_out flag)

-ped-file [file path] - The location of the phased PED file read by GERMLINE2 above.

FISHR Parameters

-reduced [integer] [numeric] - The first (integer) value tells FISHR that all final outputed segments must be at least this many SNPs in length. The second (numeric) value tells FISHR that all final outputed segments must be at least this long in cM.

-window [integer] The length, in SNPs, of the moving window to calculate the moving average of IEs.

`-emp_ma_threshold` [numeric] The maximum value for the proportion of implied errors within a window of SNPs before a segment is terminated. A lower value (e.g., .04) will result in shorter segments being discovered, with a higher likelihood that the ends are truly IBD but often under-extended, while a higher value (e.g., .08) will result in longer segments where the endpoints may be over-extended.

`-emp_pie_threshold` [numeric] The maximum proportion of implied errors that are allowed in a final called segment. A large value (e.g., .04) will allow more errors, and thus, potentially result in segments that are not truly IBD but a higher sensitivity. A smaller value (e.g., .01) will be more conservative, allowing fewer errors and resulting in segments that are more likely to be IBD but at the cost of more false negatives.

`-gap` [integer] The maximum distance, in SNPs, between two called IBD candidate segments from GERMLINE2 that are combined into one segment. Note that, if evidence so indicates, these can be broken apart again by the moving average procedure.

`-count_gap_errors` [TRUE/FALSE] - This parameter is either TRUE or FALSE. If TRUE, it will count the SNP(s) from the gap argument as errors when calculating the `-emp_ma_threshold` and the `-emp_pie_threshold`.

`-output_type` [character] - This determines which one of several different output files are generated. *finalOutput* is the default and the one most users should use. Other options are *finalErrorsOutput* (as *finalOutput*, except adds a final column that is PIE of each SH), *Reduced* (like *finalOutput* except creates smaller files by omitting 3 columns), *FullPlusDropped* (like *finalErrorsOutput* but also includes the SH's that were dropped and the reason for the drop), *Error1* (outputs error information for all the SHs that existed in the original GERMLINE2 file after consolidation, and includes three additional columns: start/stop place (in SNPs) for SH; location of each error, separated by "/"; and the PIE for each SH), and *Error2* (outputs error information for the final SHs; the number of rows for this file will therefore be the same as that for *finalOutput*).

`-log-file` [path] - File where the log information is written.

Procedure

First run GERMLINE2 for finding candidate segments for FISHR and then run FISHR, which will help remove false positives and more accurately identify segment endpoints. `Note again:` the PED file input to both GERMILNE2 and FISHR must be phased for this to work.

GERMLINE2 Example :
GERMLINE2 `-pedfile` Test.8k.ped `-mapfile` Test.8k.map `-outfile` Fin.8k `-bin_out -err_hom` 1 `-err_het` 1 `-reduced -bits` 60 `-min_m` 1.5 `-w_extend -h_extend`

You should have identified 2,343,583 IBD segments; some are false positives (which will hopefully be dropped by FISHR) and hopefully few enough are false negatives (because we can't recover those at this point). This is one reason why the thresholds above are a bit more lenient than what we'd use if we used GERMLINE2 alone.

FISHR Example :
FISHR `-bmatch` Fin.8k.bmatch `-bsid` Fin.8k.bsid `-bmid` Fin.8k.bmid `-reduced` 64 3 `-ped-file` Test.8k.ped `-window` 50 `-empirical-ma-threshold` .045 `-gap` 30 `-emp-pie-threshold` .015 `-count.gap.errors` TRUE `-output-type` finalOutput `-log-file` Fin.8kLOG `|` gzip `-c >` Fin.8k.gz

Output Files

Note that FISHR writes to standard out, so needs to be piped to (e.g., gzip) and written to file. Look at Fin.8kLOG.log (less Fin.8kLOG.log). You see the number of consolidations (two SHs put together) was 31716, # removed due to initial length not being > 3 cM (after consolidation) was $\sim$ 1.79M, #

removed due to the length following the MA trimming < 3 cM ~ 1.83M,# of those trimmed segments which still have a PIE $> .015$ being 739 and those were removed, and total number of outputted SHs > 3cM passing all thresholds is 483217.

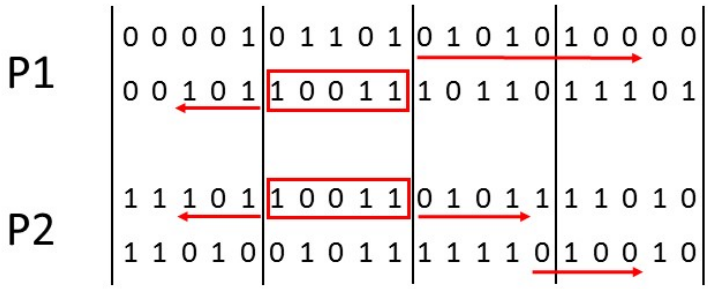

Figure 1: This figure displays the GERMLINE2 process of finding a potential segment and the extension process. The chromosomal segment is divided into smaller segments with a length defined by the `-bits` argument (in this case `-bits` 5) as denoted by the vertical lines. The perfect match in the smaller segments is denoted by the square around the variants. Then, the segment is extended until an opposite homozygous SNP is detected (if using only the `-w_extend` flag in GERMLINE2) or until enough phased mismatches occur (if using both the `-w_extend` and `-h_extend` flags).

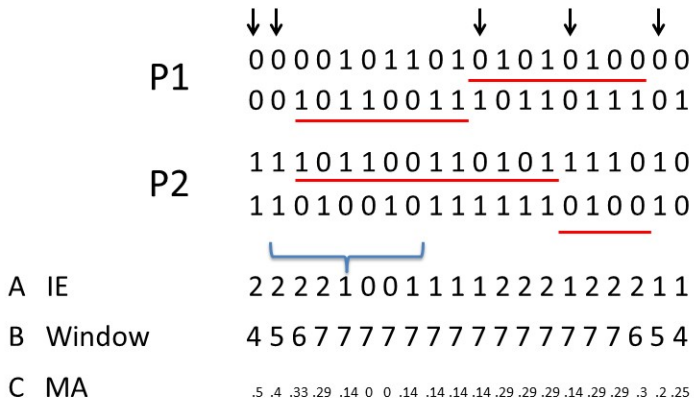

Figure 2: The moving average of IEs (MA) is calculated for each SNP within and around the called segment. In this example the `-window` value is 7, each SNP will use that SNP, as well as the three SNPs on either side of it, to calculate the MA value for that particular SNP.

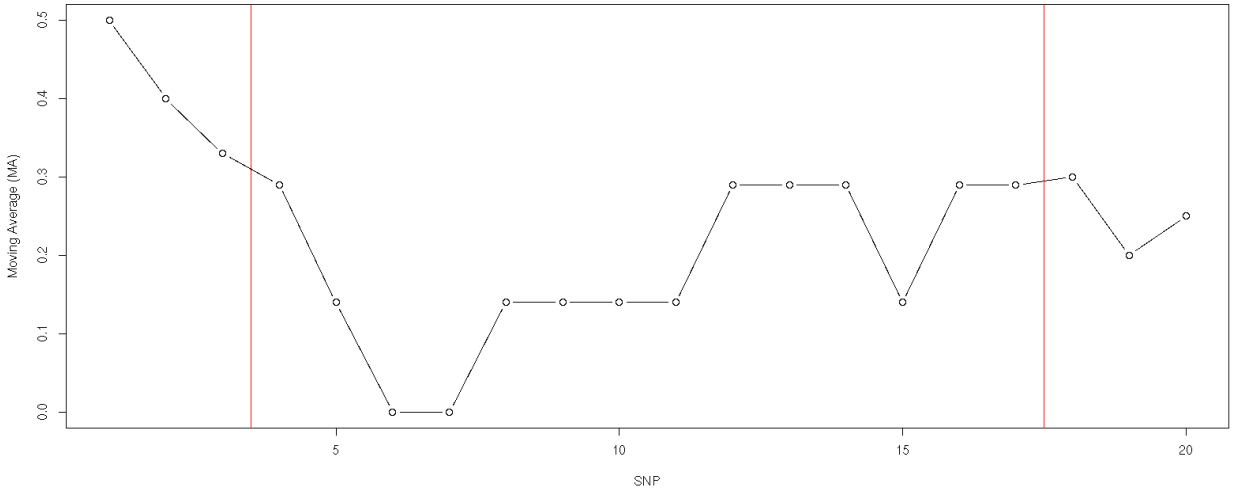

Figure 3: Once all the MA values are calculated, FISHR then starts in the center of the segment and move outward until a value is reached that is greater than the `-emp_ma_threshold` value. The point at which the MA for a SNP is above this value will determine the end points of the called segment. In this example the `-emp_ma_threshold` is 0.3 and the IBD segment will be called with the endpoints at SNP 4 and 17.

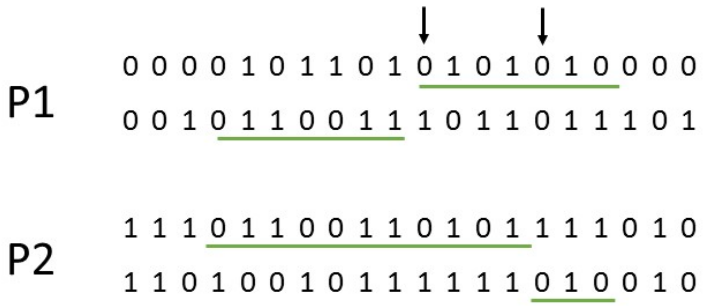

Figure 4: Once the endpoints are determined using the `-emp_ma_threshold`, the total number of IEs is calculated. If this value is greater than that provided by the `-emp_pie_threshold`, the entire segment will be removed.