GARLIC - Genomic Autozygosity Regions Likelihood-based Inference and Classification v1.1.4 User Manual

Zachary A Szpiech

May 16, 2017

Contents

1 Introduction

Extended tracts of homozygosity in individual genomes manifest as a result of haplotypes inherited identical by descent (IBD) from both biological parents. These runs of homozygosity (ROH) are important genomic features, with their length distributions and genomic locations informative about population history and useful for the mapping of recessive loci contributing to Mendelian and complex diseases. However, precisely because the length distribution of ROH is affected by population history, a simple one-size-fits-all genotype-counting approach to inferring ROH in multiple populations is ill-advised. Here, we present a model-based method and accompanying software package for inferring ROH in microarray-derived SNP genotype that incorporates population-specific parameters and genotyping error rates as described in Pemberton *et al.* (2012). Version 1.1.0 includes several updates, including a method for reweighting LOD scores by gaps between SNPs, the option to report ROH lengths in cM, and the ability to provide per-genotype error rates.

If you use GARLIC please cite the following articles,
ZA Szpiech, A Blant, TJ Pemberton. (2017) GARLIC: Genomic Autozygosity Regions Likelihood-based Inference and Classification. Bioinformatics doi: 10.1093/bioinformatics/btx102.

TJ Pemberton, et al. (2012) Genomic patterns of homozygosity in worldwide human populations. American Journal of Human Genetics, 91, 275 – 292.

2 Obtaining `GARLIC`

NOTE: In order to successfully run, windows users must have ann_figtree_version.dll and figtree.dll (provided) in the same folder as garlic.exe.

`GARLIC` pre-built binaries and source code are available at `https://github.com/szpiech/garlic`. Binaries have been compiled on OSX 10.8.5, Ubuntu 12.04 (LTS), and Windows 7, but they should function across most versions of these operating systems. To compile from source, change directories to the `src/` directory and type `make`. Some minor modification (commenting and uncommenting certain lines) to the Makefile may be necessary depending on your target OS. `selscan` depends on the zlib library (`http://www.zlib.net/`), GNU GSL (`http://www.gnu.org/software/gsl/gsl.html`), and FIGTree (`https://github.com/vmorariu/figtree`). A win32 implementation of zlib is available at `http://gnuwin32.sourceforge.net/packages/zlib.htm`. The windows version of `GARLIC` was built using a MinGW environment (`http://www.mingw.org/`), although it should only be necessary to set this environment up if you wish to compile from source on Windows. Precompiled libraries for each OS are included in the source code.

3 Basic Usage

GARLIC can be run by executing the following command.

```
garlic --tped example.tped.gz --tfam example.tfam --build hg18 --error 0.001 --auto-winsize
--out example
```

The example data is derived from human genotype chip data with hg18 coordinates (`--build hg18`), and a priori we do not know how large to make our window so we use the built-in window size selection algorithm (`--auto-winsize`). All output files will be named starting with 'example' (`--out example`). This produces the following files:

```
example.error
example.log
example.60SNPs.kde
example.freq.gz
example.roh.bed
```

If the automatic window size selection algorithm fails, you can output the KDEs of the LOD score distribution for multiplw window sizes (without calling ROH) by using the `--winsize-multi` argument, i.e.

```
garlic --tped example.tped.gz --tfam example.tfam --build hg18 --error 0.001 --winsize-multi
30 40 50 60 70 80 90 --out example
```

This will generate KDEs for your inspection. Once you've chosen a window size, you should rerun garlic specifying that size. For example, if you choose a window size of 60 SNPs, then you would run

```
garlic --tped example.tped.gz --tfam example.tfam --build hg18 --error 0.001 --winsize 60
--out example
```

If you already know what LOD score cutoff to use (say you are analyzing more individuals from a previously studied population), you can use the `--lod-cutoff` argument, i.e. if your LOD score cutoff is known to be 2.5 then you would run

```
garlic --tped example.tped.gz --tfam example.tfam --build hg18 --error 0.001 --winsize 60
--out example --lod-cutoff 2.5
```

If you already know what size thresholds to use for size classificaation (say you are analyzing more individuals from a previously studied population), you can use the `--size-bounds argument`, i.e. if your size thresholds are known to be 500000 and 1000000 for the boundaries between short/med and med/long, respectively, then you would run

```
garlic --tped example.tped.gz --tfam example.tfam --build hg18 --error 0.001 --winsize 60
--out example --lod-cutoff 2.5 --size-bounds 500000 1000000
```

To run the wLOD, you must provide a map file and give the –weighted flag:

```
garlic --tped example.tped.gz --tfam example.tfam --map example.map.gz --weighted --build
hg18 --winsize 60 --out example --error 0.001
```

Other command line arguments are described below.

4 Statistics implemented

We previously advocated the application of a logarithm of the odds (LOD) score measure of autozygosity applied in a sliding-window framework to infer ROH in high-density SNP genotype data (Pemberton *et al.*, 2012). Fundamentally, for SNP k in individual i, this method calculates the log-likelihood ratio of observed genotype $G_{i,k}$ under the hypotheses of autozygosity and non-autozygosity, incorporating an assumed genotype error rate ϵ and population-specific allele frequencies. The LOD score of window w in

individual i is then calculated as the sum of the log-likelihood ratios of the K SNPs in the window:

$$LOD(w,i) = \sum_{k=1}^{K} \log_{10}\Big(\frac{Pr[G_{i,k}|X_k=1]}{Pr[G_{i,k}|X_k=0]}\Big). \tag{1}$$

Here, $Pr[G_{i,k}|X_k=1]$ is the probability of observing genotype $G_{i,k}$ under the hypothesis of autozygosity ($X_k=1$), and $Pr[G_{i,k}|X_k=0]$ is the probability of observing genotype $G_{i,k}$ under the hypothesis of non-autozygosity ($X_k=0$). For a biallelic locus with alleles A and B that have population frequencies p_A and p_B and a genotype error rate ϵ, the genotype probabilities under the autozygosity and non-autozygosity hypotheses are given in Table 1.

Calculating $LOD(w,i)$ for all windows in all individuals in a given sample set, examination of the distribution of scores shows clear bimodality, with windows in the left-hand mode supporting the hypothesis of non-autozygosity and those in the right-hand mode supporting the hypothesis of autozygosity. The area under the autozygous mode decreases with increasing window size until it disappears, potentially reflecting the window size beyond which window length is often longer than ROH length leading to the inclusion of non-autozygous regions in the $LOD(w,i)$ calculation that mask the presence of autozygosity. A logical window size to use for ROH detection is therefore the largest window size where the distribution of LOD(w,i) is bimodal, with windows defined as autozygous if their $LOD(w,i)$ is greater than the local minimum between the two modes. Contiguous autozygous windows are subsequently joined to define ROH.

If the weighted flag is used, wLOD scores will be calculated as

$$wLOD(w,i) = \sum_{k=1}^{K} \log_{10}\Big(\frac{Pr[G_{i,k}|X_k=1]}{Pr[G_{i,k}|X_k=0]}\Big) \times Corr(p_k,[p_1,p_K]) \tag{2}$$

$$\times\, Pr[norecombination|[g_{k-1},g_k]] \tag{3}$$

$$\times\, Pr[nomutation|\mu,[p_{k-1},p_k]], \tag{4}$$

where

$$Corr(p_k,[p_1,p_K]) = \frac{1}{\sum_1^K LD_{k,i}}, \tag{5}$$

$$Pr[norecombination|[g_{k-1},g_k]] = e^{-2M(g_k-g_{k-1})}, \tag{6}$$

and

$$Pr[nomutation|\mu,[p_{k-1},p_k]] = e^{-2M\mu(p_k-p_{k-1})}. \tag{7}$$

$LD_{k,i}$ is computed as r^2 if data are phased and as HR^2 if data are unphased. g_i and p_i are the genetic distance and physical distance at marker i, respectively.

Table 1: Probability model for genotypes under autozygosity and non-autozygosity.

Observed genotype G_k	$Pr[G_{j,k}\|X_k=1]$	$Pr[G_{j,k}\|X_k=0]$
AA	$(1-\epsilon)p_A+\epsilon p_A^2$	p_A^2
AB	$2\epsilon p_A p_B$	$2p_A p_B$
BB	$(1-\epsilon)p_B+\epsilon p_B^2$	p_B^2
Missing	1	1

5 Program Options

Using the command line flag `--help`, will print a help dialog with a summary of each command line option.

5.1 Input Files

All genetic data is required to be in TPED/TFAM format (see `http://pngu.mgh.harvard.edu/~purcell/plink/data.shtml#tr`) and may be directly read as a gzipped (`http://www.gzip.org/`) version without first decompressing. Consecutive loci are assumed to be in order with respect to their physical location on the chromosome. `garlic` assumes only one population per file, and different populations should be computed separately. VCF support is planned, but until then the companion perl script `vcf2tped.pl` will convert VCF files to TPED/TFAM files.

5.1.1 File formats

tped/tfam (required):

`http://zzz.bwh.harvard.edu/plink/data.shtml#tr`

map (required when –weighted or –cm are set):

```
<chr> <snpid> <genetic pos> <physical pos>
http://zzz.bwh.harvard.edu/plink/data.shtml#map
(negative positions will cause errors)
```

tgls (optional):

```
<chr> <snpid> <unused> <physical pos> <GL ind1> <GL ind2> ... <GL indN>
```

freq (optional, output during computation, can be generated without running whole pipeling with –freq-only):

```
CHR SNP POS ALLELE FREQ
<chr> <locus ID> <allele> <freq>
```

centromere file (required if no –build is set):

```
<chr> <centromere start> <centromere end>
```

5.1.2 `--tped`

Use `--tped <string>` to specify a .tped (transposed PLINK; Purcell *et al.* (2007)) file (see `http://zzz.bwh.harvard.edu/plink/data.shtml#tr` for exact specifications) containing genetic variant information.

5.1.3 `--tped-missing`

Use `--tped-missing <char>` to specify the missing data code in the .tped file. Default is `0`.

5.1.4 `--tfam`

Use `--tfam <string>` to specify a .tfam (transposed PLINK; Purcell *et al.* (2007)) file (see `http://zzz.bwh.harvard.edu/plink/data.shtml#tr` for exact specifications) containing sample information.

5.1.5 --tgls

Use `--tgls <string>` A tgls file containing per-genotype likelihoods. Formatted: ¡chr¿ ¡snpid¿ ¡unused¿ ¡physical pos¿ ¡GL ind1¿ ¡GL ind2¿ ... ¡GL indN¿. Default: none.

Per-genotype likelihoods should be given as either a phred-scaled (`--gl-type PL`) or a $\log_{10}$-scaled (`--gl-type GL`) representing the probability that the genotype is correct or a phred-scaled likelihood that the genotype is wrong (`--gl-type GQ`). For example, if $p = 0.999$ is the probability that the genotype is correct, then $PL = -10\log_{10}(p) = 0.00434511774018$, $GL = \log_{10}(p) = -0.000434511774018$, and $GQ = -10\log_{10}(1-p) = 30$.

5.1.6 --centromere

Use `--centromere <string>` to specify a file containing custom centromere boundaries formatted as

```
<chr> <start> <end>
```

5.1.7 --freq-file

`--freq-file <string>` A file specifying allele frequencies for all variants. File format:

```
CHR     SNP     POS     ALLELE  FREQ
<chr>   <locus ID>      <pos>   <allele>        <freq>
```

By default, this is calculated automatically from the provided data.

5.1.8 --map

`--map <string>` Provide a scaffold genetic map, sites that aren't present within this file are interpolated. Sites outside the bounds are filtered. This is required for wLOD calcualtions and any runs for which you wish to report ROH in units of cM. Formatted: ¡chr¿ ¡snpid¿ ¡genetic pos¿ ¡physical pos¿. http://zzz.bwh.harvard.edu/plink/data.shtml#map (negative positions will cause errors)

5.2 Output Files

`garlic` produces several files as output. The `.log` file will record the runtime parameters and `.error` will record any errors that occur.

A gzipped `.freq` file will be output giving the allele frequencies for each locus in the dataset. Formatted as

```
CHR     SNP     POS     ALLELE  FREQ
<chr>   <locus ID>      <pos>   <allele>        <freq>
```

Kernel density estimation of the LOD score distribution is output to `.<winsize>SNPs.kde` formatted as

```
<LOD score> <density>
```

ROH calls are output to `.roh.bed`, which are formatted as UCSC BED files. There is one track per individual (all individuals in the same file) and the data are formatted as

```
<chr> <ROH start> <ROH end> <size class> <ROH length> <placeholder> <
   placeholder> <placeholder> <RGB track color>
```

When requested, raw LOD scores are also output. This file contains one row per individual (corresponding to the order in the `.tfam` file), and LOD scores for each window are given on the columns.

`<LOD score window 1> <LOD score window 2> ... <LOD score window N>`

5.2.1 --out

Use `--out <string>` to provide a base name for an output file. This will be used in place of `<outfile>` above. Default value is `outfile`.

5.3 Controlling GARLIC

5.3.1 --M

`--M <int>` The expected number of meioses since a recent common ancestor for `--weighted` calculations. Default: 7.

5.3.2 --auto-overlap-frac

`--auto-overlap-frac <bool>` If set, GARLIC attempts to guess based on marker density. Default: false.

5.3.3 --auto-winsize

`--auto-winsize <bool>` If `--weighted` is set, guesses the best window size based on SNP density, otherwise initiates an ad hoc method for automatically selecting the # of SNPs in which to calculate LOD scores. Starts at the value specified by `--winsize` and increases by ¡step size¿ SNPs until finished. Default: false.

5.3.4 --auto-winsize-size

`--auto-winsize-step <bool>` Step size for auto winsize algorithm. Default: 10.

5.3.5 --build

`--build <string>` Choose which genome build to use for centromere locations (hg18, hg19, or hg38). Default: none. A custom centromere boundary file can be passed with `--centromere`.

5.3.6 --cm

`--cm <bool>` Construct ROH in genetic distance units. This requires a mapfile.

5.3.7 --error

`--error <double>` The assumed genotyping error rate.

5.3.8 --freq-only

`--freq-only <bool>` If set, calculates a freq file from provided data and then exits. Uses minimal RAM. Default: false.

5.3.9 --gl-type

`--gl-type <string>` Specify the form of the genotype likelihood data: GQ, GL, PL, as defined in VCFv4.2 documentation. Default: none.

Per-genotype likelihoods should be given as either a phred-scaled (`--gl-type PL`) or a $\log_{10}$-scaled (`--gl-type GL`) representing the probability that the genotype is correct or a phred-scaled likelihood that the genotype is wrong (`--gl-type GQ`). For example, if $p = 0.999$ is the probability that the genotype is correct, then $PL = -10\log_{10}(p) = 0.00434511774018$, $GL = \log_{10}(p) = -0.000434511774018$, and $GQ = -10\log_{10}(1-p) = 30$.

5.3.10 --kde-subsample

`--kde-subsample <int>` The number of individuals to randomly sample for LOD score KDE. If there are fewer individuals in the population all are used. Set ≤ 0 to use all individuals (may use large amounts of RAM). Default: 20.

5.3.11 --lod-cutoff

`--lod-cutoff <double>` For LOD based ROH calling, specify a single LOD score cutoff above which ROH are called in all populations. By default, this is chosen automatically with KDE.

5.3.12 --max-gap

`--max-gap <int>` A LOD score window is not calculated if the gap (in bps) between two loci is greater than this value. Default: 200000.

5.3.13 --mu

`--mu <double>` Mutation rate per bp per generation for –weighted calculation. Default: 1.000000e-09.

5.3.14 --nclust

`--nclust <int>` Set number of clusters for GMM classification of ROH lengths. Default: 3.

5.3.15 --no-kde-thinning

`--no-kde-thinning <bool>` Set this flag to send all LOD score data to KDE function. This may dramatically increase runtime. Default: false.

5.3.16 --overlap-frac

`--overlap-frac <double>` The minimum fraction of overlapping windows above the LOD cutoff required to begin constructing a run. This is similar to PLINK's `--homozyg-window-threshold` option. ROH will have a lower bound size threshold of WINSIZE*OVERLAP_FRAC. If set to 0, GARLIC sets the value to the lowest sensible value: 1/winsize. Default: 0.25

5.3.17 --phased

`--phased <bool>` Set if data are phased and you want to calculate r2 instead of hr2 while –weighted is set. Uses extra RAM. Has no effect on computations without –weighted. Default: false.

5.3.18 --raw-lod

`--raw-lod <bool>` If set, LOD scores will be output to gzip compressed files. Default: false.

5.3.19 --resample

`--resample <int>` Number of resamples for estimating allele frequencies. When set to 0 (default), garlic will use allele frequencies as calculated from the data. If multiple populations with different sample sizes will be compared, it is recommended that you choose the same value for each analysis (i.e. 40).

5.3.20 --size-bounds

`--size-bounds <double1> ... <doubleN>` Specify the size class boundaries ROH boundaries. By default, this is chosen automatically with a 3-component GMM. Must provide numbers in increasing order.

5.3.21 --threads

`--threads <int>` The number of threads to spawn during weighted calculations. Default: 1.

5.3.22 --winsize

`--winsize <int>` The window size in number of SNPs in which to calculate LOD scores. ROH will have a lower bound size threshold of WINSIZE*OVERLAP_FRAC. Default: 10, which is almost certainly a poor choice for all analyses.

5.3.23 --winsize-multi

`--winsize-multi <int1> ... <intN>` Provide several window sizes (in number of SNPs) to calculate LOD scores. LOD score KDEs for each window size will be output for inspection.

6 Change Log

```
16MAY2017 - v1.1.4 If a freq file is provided, GARLIC now ensures that
   internal coding of alleles is consistent with the allele specified in
   the file. Improved handling of loci with 100% missing data. Improved
   reading of gzipped freq files.

12MAY2017 - v1.1.3 adds a new option for --gl-type called GQ, which is a
   phred-scaled probability that the genotype is wrong. If p = 0.999 is
   the probability that the genotype is correct, then GQ = -10*log10(1-p)
    = 30.  The default for --gl-type is now set to none, so if you
   provide a TGLS file you will be required to choose between GQ, PL, and
    GL.

08MAY2017 - v1.1.2b fixes a bug that causes a crash when --overlap-frac
   is set to 0. Setting to zero now chooses the smallest sensible overlap
    fraction (1/winsize).  If you wish to let garlic automatically guess
   overlap fraction, use the new command line option: --auto-overlap-frac
```

. Also includes an interation counter to monitor progress of GMM portion of the code.

07MAY2017 - v1.1.2 fixes a bug that causes a crash when a tgls file indicates a genotype has 0 likelihood. Add progress bars for LD calculations and LOD score calculations. Improve efficiency of tgls file loading.

06MAY2017 - v1.1.1a fixes a bug introduced with 1.1.1 that caused crashed when --resample is used.

05MAY2017 - Update to version 1.1.1. For some large datasets (e.g. WGS), KDE can take extremely long to complete, even when only a subset of individuals are considered. Here we introduce some changes to speed this calculation up. Instead of passing all LOD score windows to the KDE function, we now pass only non-overlapping windows. This reduces the number of points sent to the function by a factor of WINSIZE. If you wish to retain all LOD scores, you may set the --no-kde-thinning flag. We also change the default number of random individuals for KDE (--kde-subsample) to 20, which still may be set to 0 to use all individuals.

We also introduce a lower bound on the size of ROH reported. Now, an ROH will not be called unless it contains at least WINSIZE*OVERLAP_FRAC number of SNPs. For example, for a WINSIZE (--winsize) of 100 and an OVERLAP_FRAC (--overlap-frac) of 0.25 the smallest possible ROH that will be reported will be 25 SNPs long.

Finally, if --overlap-frac is set to 0, GARLIC will attempt to guess a good choice.

02MAY2017 - Update to version 1.1.0a. Bug fixed that caused crashes when using TGLS file.

28APR2017 - Update to version 1.1.0. Includes several updates, including a method for reweighting LOD scores by gaps between SNPs, the option to report ROH lengths in cM, and the ability to provide per-genotype error rates.

25OCT2016 - Update to version 1.0.1. This update fixes a bug where ROH that extended upto the end of the chromosome failed to get assembled and reported. This update also introduces a new command-line flag --overlap-frac which is designed to reduce false positive calls by requiring a SNP be covered by at least OVERLAP_FRAC (default 0.25) proportion of high scoring windows to be included in an ROH call. This helps to reduce false positive calls near the boundaries of true ROH.

05MAY2016 - Initial release of GARLIC v1.0.0

References

Pemberton, T. J., Absher, D., Feldman, M. W., Myers, R. M., Rosenberg, N. A., and Li, J. Z. 2012. Genomic patterns of homozygosity in worldwide human populations. *American Journal of Human Genetics*, 91: 275–292.

Purcell, S., Neale, B., K., T.-B., Thomas, L., Ferreira, M. A. R., Bender, D., Maller, J., Sklar, P., de Bakker, P. I. W., Daly, M. J., and Sham, P. C. 2007. PLINK: a toolset for whole-genome association and population-based linkage analysis. *American Journal of Human Genetics*, 81: 559–575.