BEAGLECALL 1.0

Brian L. Browning

Department of Medicine

Division of Medical Genetics

University of Washington

15 November 2010

Contents

1 Introduction

BEAGLECALL is a software program for calling genotypes and inferring haplotypes from normalized allele signal intensity data. BEAGLECALL incorporates the BEAGLE haplotype frequency model, and this enables BEAGLECALL to achieve higher accuracy than genotype calling methods that do not make use of linkage disequilibrium.

BEAGLECALL is written in Java and runs on most computing platforms (including Windows, Unix, Linux, Solaris, and Mac). A Java interpreter is probably already installed on your computer (type java -version at the command line prompt to check). However, if it is not installed or if it is not version 1.6 or later, the up-to-date version of the Java interpreter can be downloaded free of charge from the **java.sun.com** web site. The Java interpreter is called the Java Standard Edition (SE) Runtime Environment (JRE).

1.1 Citing BEAGLECALL

If you use BEAGLECALL and publish your analysis, please report the version of the program used, and please cite the following article.

> B L Browning and Z Yu (2009) Simultaneous genotype calling and haplotype phasing improves genotype accuracy and reduces false positive associations for genome-wide association studies. The American Journal of Human Genetics 85:847-861. doi:10.1016/j.ajhg.2009.11.004

1.2 Files in the BEAGLECALL software distribution

BEAGLECALL is freely available and can be downloaded from the BEAGLECALL web site:

http://faculty.washington.edu/browning/beaglecall/beaglecall.html

The following files are available:

1. beaglecall_1.0_[date].pdf - the BEAGLECALL 1.0 documentation.
2. beaglecall.jar - the BEAGLECALL 1.0 executable file.
3. beaglecall_example.zip - a folder with example data for 500 individuals and 99 markers. The example data is a subset of the ILLUMINUS[1] example data (see Acknowledgements in Section 6).
4. beaglecall_[version]_[date].src.zip - source code for BEAGLECALL (does not include source code for BEAGLE).

Utility programs that can be used to prepare input files and process output files are available at the BEAGLE utilities web site:

http://faculty.washington.edu/browning/beagle_utilities/utilities.html

There are BEAGLE utilities for extracting lines or columns from a file, for pasting files together, for transposing rows and columns of a file, for converting genotype probabilities to called genotypes, and for converting between linkage format and BEAGLE format. Standard Unix utilities, such as cat, zcat, head, tail, tr, cut, sort, uniq, and wc, are also useful when working with text files.

2 File formats

BEAGLECALL imposes very few constraints on your data files. Data fields on each line can be separated by a space, a tab, or any combination of white-space characters. Allele and marker identifiers can be any sequence of characters that does not contain white space. Input files can be compressed with the GZIP algorithm. BEAGLECALL assumes that any file with a name ending in “.gz” is GZIP compressed.

BEAGLECALL requires two types of input files: allele signals files and genotype probabilities files. The allele signal files contain normalized A-allele and B-allele signal intensities for each genotype. The genotype probabilities file contains current estimated genotype probabilities for each individual. The initial genotype probabilities files are typically constructed using genotype calls from another genotype calling program (see Section 3.1).

2.1 Allele signals files (.signals)

The allele signals files have a simple format: there is one row per markers and there are two columns per individual. The following example allele signals file has two individuals and three genotyped markers:

Example 1- Sample allele signals file

marker	alleleA	alleleB	1001	1001	1002	1002
rs2289311	A	G	1.502559	0.906223	0.051661	1.806582
rs1248628	C	T	1.152608	1.103111	0.178081	1.541044
rs10762764	G	T	0.768067	0.981742	1.668715	0.526407

The first line of an allele signals file is a header line that gives the sample identifier corresponding to each column of data. The first column (**marker**) lists the marker identifiers. The second and third columns (**alleleA** and **alleleB**) give the two alleles for each marker. The remaining columns give the normalized A-allele and B-allele signal intensities (in that order) for each individual: columns 4-5 give the normalized allele signal intensities for the first individual, columns 6-7 give the normalized allele signal intensities for the second individual, and so on. The two columns for an individual must have the same sample identifier in the header line. The markers in an allele signals file must be in chromosomal order.

In Example 1, the two sample identifiers are 1001 and 1002, the A-allele signal intensity for marker rs2289311 and sample 1001 is 1.502559, and the B-allele signal intensity for marker rs10762764 and sample 1002 is 0.526407.

2.2 Genotype probabilities files (.gprobs)

Genotype probabilities files have a format that is similar to the allele signals file format. The major difference is that a genotype probabilities file has three columns per individual and an allele signals file has two columns per individual. The following example genotype probabilities file has two individuals and three genotyped markers:

Example 2 - Sample genotype probabilities file

marker	alleleA	alleleB	1001	1001	1001	1002	1002	1002
rs2289311	A	G	0.0000	0.9957	0.0043	0.0000	0.0201	0.9799
rs1248628	C	T	0.0000	0.0169	0.9831	0.0469	0.9531	0.0000
rs10762764	G	T	0.0000	0.9732	0.0268	1.0000	0.0000	0.0000

The first line of a genotype probabilities file is a header line that gives the sample identifier corresponding to each column of data. The first column (**marker**) lists the marker identifiers. The second and third columns (**alleleA** and **alleleB**) give the two alleles for each marker. The remaining columns give the genotype probabilities of the AA, AB, and BB genotypes (in that order) for each individual: columns 4-6 give the genotype probabilities for the first individual, columns 7-9 give the genotype probabilities for the second individual, and so on. The markers in the genotype probabilities file must be in chromosomal order. The three columns for each individual must have the same sample identifier in the header line. The three genotype probabilities must be non-negative, but do not need to sum to 1. BEAGLECALL will scale the three probabilities to sum to 1 if they do not sum to 1.

In Example 2, the two sample identifiers are 1001 and 1002, the probability of the heterozygous genotype (AG) for marker rs2289311 and sample 1001 is 0.9957, and the probability of the homozygous G-allele genotype (GG) for marker rs10762764 and sample 1002 is 1.0000.

2.3 Genotype likelihoods files (.like)

Genotype likelihoods files convey the relative evidence for each genotype call provided by the allele signal intensity data. Genotype likelihoods files are generated and used internally by BEAGLECALL, but generally are not used for downstream analysis because the likelihoods are estimated using allele signal intensities only, without using linkage disequilibrium. The genotype likelihoods file format is the same as the genotype probabilities file format, except that a genotype likelihoods file contains genotype likelihoods instead of genotype probabilities. Thus for a genotype likelihoods file, the three columns for each individual give the estimated value of the probability density function of the observed allele signal data when the true genotype is the AA, AB, and BB genotype (in that order). The three genotype likelihoods must be non-negative, but do not need to sum to 1 since it is the ratio of likelihoods rather than their absolute value that is important. BEAGLECALL will produce an interim genotype likelihood file corresponding to each input allele signals file (see Section 5.8). When the default `runbeagle=true` option is used (see Section 4.3), BEAGLECALL will use the interim likelihood files and the BEAGLE program to generate posterior genotype probabilities.

3 Running BEAGLECALL

3.1 Preparing input files

First, the array probe data must be normalized to reduce non-biological sources of variation, and the probe data for each genotype must be summarized by an A-allele and a B-allele signal intensity.[2]

Second, if there are systematic differences in the normalized allele signal intensity data between subsets of your sample (e.g. due to differences in DNA collection, processing, storage, or genotyping), then you should divide your sample into cohorts to minimize the within-cohort differences in allele signal intensity data (i.e. batch effects) and place data for each cohort in a separate allele signals file. The format for an allele signals file is described in Section 2.1. BEAGLECALL models allele signal intensity data in each input file separately and then combines the results when estimating linkage disequilibrium. So you must ensure that the number of samples in a cohort is large enough to adequately model the allele signal intensity data for the cohort. You must also ensure that all allele signal data in an allele signals file was normalized together. If you use the normalized allele signal intensity data from an outside source, such as dbGaP, you should discover which samples were normalized together, and you should not include samples that were normalized separately in the same allele signals file.

Third, create an initial genotype probabilities file for each cohort. The format for a genotype probabilities file is described in Section 2.2. The initial genotype probabilities files give the initial estimates of genotype probabilities for each sample for each marker. The output of some genotype calling programs such as CHIAMO[3] and ILLUMINUS[1] include estimated genotype probabilities. If genotype probabilities from another calling program are available, you can use these genotype probabilities to create your initial genotype probabilities files.

If genotype probabilities are not available, you can use genotype calls made using another calling program (e.g. both Affymetrix and Illumina provide genotype calling software for their arrays). In this case, you will need to code the existing genotype calls as genotype probabilities. In a genotype probabilities file, each genotype has three genotype probabilities, representing the estimated probability that the true genotype is AA, AB, and BB. Thus for the initial genotype probabilities file, you should code AA genotypes as “1 0 0”, AB genotypes as “0 1 0”, BB genotypes as “0 0 1”, and missing genotypes as “0.333 0.333 0.333”. There is a BEAGLE utility called bgl2gprobs.jar that converts called genotypes to genotype probabilities (see Section 1.2).

Fourth, identify any samples which have poor quality data and should be excluded. For example, if you have genotype calls made with the vendor’s software you may want to exclude samples with unusually low genotype call rates, or with unusually high or low numbers of heterozygote genotypes. Place the sample identifiers in a text file with one sample identifier per line. When you run BEAGLECALL, the file with excluded sample identifiers will be specified with the excludesamples command line argument described in Section 4.1.

You do not need to manually identify markers to be excluded because BEAGLECALL has built-in data quality filters that can exclude markers with poor quality data (see Section 3.3). However, you can also force BEAGLECALL to exclude specific markers with the excludemarkers command line argument described in Section 4.1.

Typically you will want to prepare separate input files for each autosome, so that the analysis can be parallelized by chromosome. The current version of BEAGLECALL can be used to call X-chromosome genotypes in female samples, but does not support calling of X-chromosome or Y-chromosome data in male samples.

Data Consistency. BEAGLECALL ignores all data for excluded samples and excluded markers when checking data consistency. After ignoring excluded samples in a cohort, the remaining samples in the allele signals file and genotype probabilities file must be the same and in the same order. After excluding the excluded markers, BEAGLECALL will check the remaining markers for consistency:

1. The markers in the input allele signals files must be the same and in the same order for all cohorts.
2. The markers in the input genotype probabilities files must be the same and in the same order for all cohorts.
3. The markers in the genotype probabilities files must be either the same as or a subset of the markers in the allele signals files. The order of the markers in the genotype probabilities files and allele signals files must be consistent.

3.2 Running BEAGLECALL in a UNIX environment

This section gives an example of the genotype calling process for a Unix or Linux environment. For this example we will assume you are calling genotypes for markers on chromosome 1 genotyped on an Affymetrix array. We will assume that there are batch effects that cause systematic differences between the case and control allele signal intensities. Since there are batch effects, you will use separate input files for the case and the control data. We assume that you have prepared:

1. An input allele signals file for each cohort that contains the normalized allele signal intensity data (case.chr1.signals.gz and control.chr1.signals.gz)
2. An input genotype probabilities file for each cohort (see Section 3.1) that contains current estimates of genotype probabilities (init.case.chr1.gprobs.gz and init.control.chr1.gprobs.gz).
3. An input file of samples to be excluded (sample.excl) that contains one sample identifier per line. The excluded samples file is optional.

For the first iteration, enter the following command:

```
java -Xmx1000m -jar beaglecall.jar  \
   signals=case.chr1.signals.gz  \
   signals=control.chr1.signals.gz  \
   gprobs=init.case.chr1.gprobs.gz  \
   gprobs=init.control.chr1.gprobs.gz  \
   excludesamples=samples.excl  \
   callthreshold=0.8  \
   missingcohort=0.2  \
   out=bc1.chr1  \
   &> screen.bc1.chr1 &
```

The command line arguments specify the input files, the missing data filter, the output file prefix (bc1.chr1), and a file to receive the screen output (screen.bc1.chr1). The use of the “\” immediately before the (invisible) end-of-line character permits the command to be split over several lines. The order of the cohorts (e.g. cases first, controls second in this example) must be the same for the input allele signals files and the input genotype probabilities files. The callthreshold=0.8 and missingcohort=0.2 arguments specify the missing data filter. In the first iteration, the missing data filter will exclude any marker with >20% of samples having maximum genotype probability <0.8 in one or more of the input genotype probabilities files. See Section 3.3 for details of BEAGLECALL's built in data quality filters.

A number of output files are produced, all of which begin with the prefix "bc1.chr1." We include a "bc1" in our output file names to indicate that the output files are from the first BEAGLECALL iteration. The most important output files are the log file (bc1.chr1.log) that summarizes the analysis and the two output genotype probabilities files which give updated genotype probability estimates for the cases (bc1.chr1.case.chr1.signals.gz.like.gz.gprobs.gz) and for the controls (bc1.chr1.control.chr1.signals.gz.like.gz.gprobs.gz). The filenames for the output genotype probabilities files are obtained by adding the output prefix "bc1.chr1." and the suffix ".like.gz.gprobs.gz" to the name of the allele signals file for the cohort (case.chr1.signals.gz and control.chr1.signals.gz).

The -Xmx1000m command line argument tells the Java interpreter to allocate up to 1000 megabytes of memory for this analysis. In general, the amount of memory required for an analysis will depend on the sample size (see Section 4 for more details).

For the second iteration, enter the following command:

```
java -Xmx1000m -jar beaglecall.jar  \
   signals=case.chr1.signals.gz  \
   signals=control.chr1.signals.gz  \
   gprobs=bc1.chr1.case.chr1.signals.gz.like.gz.gprobs.gz  \
   gprobs=bc1.chr1.control.chr1.signals.gz.like.gz.gprobs.gz  \
   excludesamples=samples.excl  \
   callthreshold=0.96 \
   missingcohort=0.04 \
   out=bc2.chr1  \
   &> screen.bc2.chr1
```

In the command for the second iteration, we use "bc2.chr1" for our output file prefix, we use the first iteration's output genotype probabilities files as our input genotype probabilities files (specified with the gprobs arguments), and we use a more stringent missing data filter. For the second iteration, the missing data filter will exclude any marker with >4% of samples having maximum genotype probability <0.96 in one or more of the input genotype probabilities files.

For the third iteration, we make the missing data filter for the input genotype probabilities files a bit more stringent (callthreshold=0.97 and missingcohort=0.03), and we increment the iteration (changing "bc1" to "bc2" and changing "bc2" to "bc3"):

```
java -Xmx1000m -jar beaglecall.jar  \
   signals=case.chr1.signals.gz  \
   signals=control.chr1.signals.gz  \
   gprobs=bc2.chr1.case.chr1.signals.gz.like.gz.gprobs.gz  \
   gprobs=bc2.chr1.control.chr1.signals.gz.like.gz.gprobs.gz  \
   excludesamples=samples.excl  \
   callthreshold=0.97 \
   missingcohort=0.03 \
   out=bc3.chr1  \
   &> screen.bc3.chr1
```

In my experience, using three BEAGLECALL iterations performs well in many situations. However, if additional accuracy is needed, you can run additional iterations, increasing the stringency of the missing data filter with each iteration. For example, you could use callthreshold=0.975 missingcohort=0.025 for the fourth iteration and callthreshold=0.98 missingcohort=0.02 for the fifth iteration. Note that the missing data filters in this example are calibrated for Affymetrix data. Suggested missing data filters for Illumina data are given in Section 3.3.2.

After the final iteration of BEAGLECALL, there will be an output genotype probabilities files for each cohort (bc3.chr1.case.*.gprobs.gz and bc3.chr1.control.*.gprobs.gz) that can be used to make genotype calls. Genotype calls can be made using the gprobs2beagle utility program from the BEAGLE Utilities web site (see Section 1.2). There will also be an output file of phased haplotypes for each cohort (bc3.chr1.case.*.phased.gz and bc3.chr1.control.*.phased.gz) that can be used in haplotypic analysis.

I usually apply the same missing data filter used at the start of the final iteration to the output genotype probabilities file at the end of the final iteration. Thus, in the preceding example, I would typically exclude any marker with >3% of samples having maximal genotype probability <0.97 before calling genotypes and performing further analysis. The BEAGLE Utilities web site (see Section 1.2) has utility programs that can calculate missing genotype rates for each marker in a genotype probabilities file (gprobsmissing.jar) and that can exclude specified markers (filterlines.jar) from a genotype probabilities file.

3.3 Using BEAGLECALL’s data quality filters

BEAGLECALL has built-in data quality filters that are applied to the input genotype probabilities files, and that can exclude markers with large deviations from Hardy-Weinberg equilibrium (HWE) or high levels of missing data. You can also perform manual data quality filtering or perform a combination of automatic and manual filtering. Manual data quality filtering is accomplished by placing marker identifiers to be excluded in a file and specifying the file with the excludemarkers command line argument (see Section 4.1). The BEAGLE Utilities web site (see Section 1.2) has programs that facilitate manual data quality filtering: the gprobshwe.jar utility program computes the HWE test p-value for each marker, and the gprobsmissing.jar utility program computes the missing data proportion for each marker.

3.3.1 Hardy-Weinberg equilibrium (HWE) filter

BEAGLECALL uses the genotype with highest probability in the input genotype probabilities files to compute the exact HWE P-values using the method of Wigginton et al.[4] By default, BEAGLECALL will exclude markers with an exact Hardy-Weinberg equilibrium P-value $<10^{-6}$ in any cohort or in the union of all cohorts. Different HWE P-value thresholds can be set by the user (see Section 4.2). Although a marker-disease association can cause deviation from HWE in cases, a 10^{-6} HWE P-value threshold is not expected to result in any appreciable loss of power for common diseases and typical sample sizes because HWE tests have relatively poor power to detect marker-disease association.

It is strongly recommended that you apply a HWE filter in the first BEAGLECALL iteration (see Appendix 2 of the reference in Section 1.1). In the example in Section 3.2, each BEAGLECALL iteration used the default HWE filter.

3.3.2 Missing Data Filter

The missing data filter has three parameters: a calling threshold parameter and a maximum missing data proportion parameter for each cohort, and a maximum missing data proportion for the union of all cohorts. The calling threshold determines which genotypes are missing. For example, if the calling threshold is 0.96, then any genotype in a genotype probabilities file with maximum probability <0.96 is considered missing. If the maximum missing data proportion parameter for a cohort is 0.04, then any marker with >4% of samples having missing genotypes in any genotype probabilities file will be excluded. If the maximum missing data proportion for the study is 0.04, then any marker with >4% missing genotypes for the entire study (the union of all samples in the genotype probabilities files) will be excluded. These three parameters are used only for applying a missing data filter, and have no other effect on the analysis.

Use of stringent missing data filters in the first BEAGELCALL iteration is not recommended. This is because the initial genotype probabilities in the first iteration are typically derived from an alternative genotype calling program, and applying stringent missing data filters to the input genotype probabilities may exclude many markers that can be called accurately with BEAGLECALL. However, it is recommended that a HWE filter be used in the initial BEAGLECALL iteration (see Section 3.3.1) because the BEAGLE haplotype frequency model assumes HWE.

In my experience, the best result are obtained when applying increasingly stringent genotype calling filters. For Affymetrix data, the following missing data filters are usually reasonable:

❖	Affymetrix iteration 1:	callthreshold=0.8	missingcohort=0.2
❖	Affymetrix iteration 2:	callthreshold=0.96	missingcohort=0.04
❖	Affymetrix iteration 3:	callthreshold=0.97	missingcohort=0.03

For Illumina data, the following missing data filters are usually reasonable:

❖	Illumina iteration 1:	callthreshold=0.9	missingcohort=0.1
❖	Illumina iteration 2:	callthreshold=0.98	missingcohort=0.02
❖	Illumina iteration 3:	callthreshold=0.985	missingcohort=0.015

More lenient missing data filters may be required if whole genome amplified DNA is used.

4 BEAGLECALL command line arguments

To run BEAGLECALL, enter the following command at the computer prompt:

```
java -Xmx[Mb]m -jar beaglecall.jar <arguments>
```

Each argument has the format **parameter=value**. There is no white-space between the **parameter** and **=** or between **=** and the **value**. The command line arguments are described below. Input file formats are described in Section 2. When I use BEAGLECALL, I create separate input files for each chromosome and run BEAGLECALL separately for each chromosome (in parallel). One can also divide large chromosomes into halves, thirds, etc. and run BEAGLECALL on each section separately.

The -Xmx[Mb]m command line arguments sets the amount of memory available to the Java interpreter to be [Mb] megabytes where [Mb] is a positive integer. It is helpful to set the maximum amount of memory somewhat higher than the minimum memory required to analyze your data because using additional memory can decrease computation time.

BEAGLECALL simultaneously infers genotype probabilities and haplotype phase. Haplotype phase inference is computationally demanding. Large sample sizes (say 4000-6000 samples) typically require 2-4 Gb of memory. If you do not have sufficient memory to analyze your entire sample, one work-around is to divide each cohort in halves (or thirds, fourths, etc.) and run genotype calling on each half (or third, fourth, etc) of the data separately.

BEAGLECALL creates temporary files in your system's default temporary-file directory. If your system's default temporary-file directory has insufficient space, you can specify an alternate temporary-file directory by replacing the initial "java" in the command line with "java -Djava.io.tmpdir=<directory>" where "<directory>" is the name of an alternate directory for storing temporary files.

4.1 Arguments for specifying files

- signals=<allele signals file> where <allele signals file> is the name of a file containing normalized allele signal intensity data (see Section 2.1). You may use multiple signals arguments. If there are systematic differences in the normalized allele signal intensity data between subsets of your sample (e.g. due to differences in DNA collection, processing, storage, or genotyping), then you should divide your sample into cohorts to minimize the within-cohort differences in allele signal intensity and you should prepare a separate allele signals file for each cohort. See section 3.1 for more details. The signals argument is required.
- gprobs=<genotype probabilities file> where <genotype probabilities file> is the name of a genotype probabilities file containing current estimates of genotype probabilities (see Section 2.2). You must have an equal number of signals and gprobs arguments. If you use multiple gprobs arguments the *k*-th signals argument and the *k*-th gprobs argument must correspond to the same cohort. The gprobs argument is required.
- out=<output file prefix> where <output file prefix> is a relative or absolute pathname giving the prefix for the output file names. The different output files are described in Section 5. The out argument is required.

- excludesamples=<excluded samples file> where <excluded samples file> is the name of file containing sample identifiers (one identifier per line) that will be excluded from the analysis and output files. Beagle will check the sample identifiers in the first line of each file specified with signals and gprobs arguments and exclude any column whose sample identifier is in the excluded samples file. The excludesamples argument is optional.
- excludemarkers=<excluded markers file> where <excluded markers file> is the name of a file containing marker identifiers (one identifier per line) that will be excluded from the analysis and output files. Markers can also be excluded by the built-in data quality filters (see Section 4.2). The excludemarkers argument is optional.

4.2 Arguments for data quality filtering

- hwecohort=<min cohort HWE P-value> where <min cohort HWE P-value> is the minimum permissible P-value from an exact test of Hardy-Weinberg equilibrium[4] when genotype calls are made from an input genotype probabilities files (see Section 3.3.1). For each marker, a HWE P-value is calculated for each input genotype probabilities file. If the HWE P-value for any input genotype probabilities file is less than the specified threshold then the marker will be excluded and the marker identifier will be written to the output excluded markers file (see Section 5.4). The hwecohort argument is optional. The default value is hwecohort=1.0e-6.
- hwestudy=<min study HWE P-value> where <min study HWE P-value> is the minimum permissible P-value from an exact test of Hardy-Weinberg equilibrium[4] for the combined sample when genotype calls are made from the input genotype probabilities files (see Section 3.3.1). Any marker with a HWE P-value less than the specified threshold in the combined sample will be excluded, and the marker identifier will be written to the output excluded markers file (see Section 5.4). The hwestudy argument is optional. The default value is hwestudy=1.0e-6.
- callthreshold=<min genotype probability> where <min genotype probability> is the calling threshold when applying a missing data filter to genotype calls made from the input genotype probabilities files (see Section 3.3.2). Any genotype with maximum genotype probability less than the specified threshold is considered missing when applying a missing data filter. The callthreshold argument is optional. The default value is callthreshold=0.0 which results in no missing data filtering.
- missingcohort=<max cohort missing proportion> where <max cohort missing proportion> is the maximum permitted proportion of missing genotypes when applying a missing data filter to genotype calls made from the input genotype probabilities files (see the preceding callthreshold argument and Section 3.3.2). For each marker, the missing data proportion is calculated for each input genotype probabilities file. If the missing data proportion for a marker in any input genotype probabilities file is greater than the specified threshold then the marker will be excluded and the marker identifier will be written to the output excluded markers file (see Section 5.4). The missingcohort argument is optional. The default value is missingcohort=1.0, which results in no missing data filtering.

- missingstudy=<max study missing proportion> where <max study missing proportion> is the maximum permitted proportion of missing genotypes when applying a missing data filter to genotype calls made from the union of all input genotype probabilities files (see the preceding callthreshold argument and Section 3.3.2). For each marker, the missing data proportion is calculated for the union of all samples in the input genotype probabilities files. If the missing data proportion for the union of all samples is greater than the specified threshold then the marker will be excluded and the marker identifier will be written to the output excluded markers file (see Section 5.4). The missingstudy argument is optional. The default value is missingstudy=1.0, which results in no missing data filtering.

4.3 Other arguments

- runbeagle=<true/false> where <true/false> is true if the BEAGLE[5; 6] program will be run automatically by BEAGLECALL and false if the BEAGLE program will be run manually by the user. The BEAGLE program must be run to produce output genotype probabilities that are estimated using both allele signal intensities and linkage disequilibrium. If runbeagle=false, then BEAGLE should be run manually using the interim genotype likelihood files produced by BEAGLECALL. A suggested command line for BEAGLE is printed in the output log file for the analysis (see Section 5.1). The runbeagle argument is optional. The default value is runbeagle=true.

- maxlr=<max likelihood ratio> where <max likelihood ratio> is the maximum permitted likelihood ratio when there are alternative genotype calls with non-zero likelihood. If the observed signal data is S and if the genotype likelihoods for genotypes g_1 and g_2 satisfy $0 < P(S|G = g_1) < P(S|G = g_2)$, and if $P(S|G = g_2)/P(S|G = g_1)$, exceeds the maximum permitted likelihood ratio then the smaller likelihood $P(S|G = g_1)$ is set to 0.0. The maxlr argument is optional. The default value is maxlr=5000. The default maximum permitted likelihood ratio is expected to give nearly optimal genotype accuracy. Running time and computational requirements increase as the maxlr parameter increases.

- df=<degrees of freedom> where <degrees of freedom> is a real number > 2.0 giving the degrees of freedom for the *t*-distrubutions used to model the signal data for each genotype cluster. The df argument is optional. The default value is df=5.

- maxit=<max iterations> where <max iterations> is a positive integer giving the maximum number of iterations that will be used when estimating parameters of the probability model for the allele signal intensity data. The maxit argument is optional. The default value is maxit=50.

- assaysuccess=<initial assay success probability> where <initial assay success probability> is the initial estimate of the probability that the assay for a genotype is successful and that the A-allele and B-allele signal intensities for the genotype are informative for the true genotype. The assaysuccess argument is optional. The default value is assaysuccess=0.997.

5 Output files

BEAGLECALL creates several output files, the most important of which are the log file that summarizes the analysis (Section 5.1) and the output genotype probabilities files (Section 5.9). The output filenames begin with the filename prefix specified by the out command line argument described in Section 4.1.

5.1 Log file [.log]

The log file gives a summary of the analysis that includes the BEAGLECALL version, the start time, a summary of any data quality filtering, a list of the command line arguments for the analysis, and the running time for the analysis. If the runbeagle parameter is true, the log file includes the log file for the BEAGLE analysis.

5.2 Hardy-Weinberg equilibrium P-value file [.hwe]

The HWE P-value file gives the exact HWE P-value for each cohort and for the union of all cohorts when genotype calls are made using the input genotype probabilities files (See Section 3.3.1). The algorithm for calculating the HWE P-value was developed by J. E. Wigginton, D. J. Cutler, and G. R. Abecasis.[4] The columns of the HWE P-value file report the marker identifier (first column), followed by the HWE P-value for each cohort in the order the cohorts are specified by the signals arguments in the BEAGLECALL command line (see Section 4.1), followed by the HWE P-value for the entire sample (last column).

5.3 Missing data proportion file [.miss]

The missing data proportion file gives the proportion of missing genotypes for each cohort and for the union of cohorts when genotype calls are made using the input genotype probabilities file. Missing genotypes are determined by the callthreshold parameter (see Section 4.2). The columns of the missing data proportion file report the marker identifier (first column), followed by the missing data proportion for each cohort in the order the cohorts are specified by the signals arguments in the BEAGLECALL command line (see Section 4.1), followed by the missing data proportion for the union of all cohorts (last column).

5.4 Excluded markers file [.markers.excl]

The excluded markers file lists the identifiers of markers in the genotype probabilities file that were manually excluded with the excludemarkers command line argument (see Section 4.1) or that were automatically excluded by the the Hardy-Weinberg equilibrium filter or the missing data filter. One marker identifier is printer on each line.

5.5 Markers file [.markers]

The markers file enumerates the markers, marker alleles, and the position of the markers in the output genotype probabilities files. The k-th marker is assigned position k. The markers file is the input markers file for the BEAGLE analysis when the runbeagle parameter is true (see Section 4.3).

5.6 Model files [.model]

The model files describes the probability model for the allele signal intensity data for each non-excluded marker in the input genotype probabilities files. A model file is created for each cohort specified with a signals argument in the BEAGLECALL command line (Section

4.1). The model file includes the estimated mean vector and variance/covariance matrix for the allele signal intensity data corresponding to each possible genotype, the estimated assay success parameter, and the number of iterations required to estimate the preceding parameters.

5.7 Interim genotype probabilities files [.tmpgp.gz]

The interim genotype probabilities are estimated using input genotype probabilities and allele signal intensity data, but are not estimated using linkage disequilibrium. An GZIP-compressed interim genotype probabilities file is created for each cohort specified with a signals argument in the BEAGLECALL command line (Section 4.1). The interim genotype probabilities files generally should not be used for downstream analysis. The interim genotype probabilities files are in genotype probabilities file format (see Section 2.2).

5.8 Interim likelihoods files [.like.gz]

An GZIP-compressed interim likelihoods file is created for each cohort specified with a signals argument in the BEAGLECALL command line (Section 4.1). Genotype calls should not be made from the genotype likelihoods in the interim likelihood files because the genotype likelihoods are estimated from allele signal intensities only, without using linkage disequilibrium. Instead the interim likelihood files should be used as input data for BEAGLE and genotype calls should be made from the output genotype probabilities produced by BEAGLE (see Section 5.9). BEAGLE is run automatically when the runbeagle parameter is true (see Section 4.3). The interim likelihoods files are in genotype likelihoods file format (see Section 2.3).

5.9 Genotype probabilities files [.gprobs.gz]

Output genotype probabilities are produced by BEAGLECALL when the runbeagle parameter is true (see Section 4.3). A GZIP-compressed genotype probabilities file is created for each cohort specified with a signals argument in the BEAGLECALL command line (Section 4.1). The output genotype probabilities files gives estimated genotype probabilities which can be used to call genotypes for downstream analysis. The output genotype probabilities file is also used as input data in the next iteration of the BEAGLECALL program. The genotype probabilities file format is described in Section 2.2.

5.10 Genotype dosage File [.dose.gz]

If a sample has genotype probabilities $(P(AA), P(AB), P(BB))$ for a marker, then the estimated B-allele dosage is $0 \times P(AA) + 1 \times P(AB) + 2 \times P(BB)$. A GZIP-compressed genotype dosage file is created for each cohort specified with a signals argument in the BEAGLECALL command line (Section 4.1). The header line of the genotype dosage file is similar to the header line of the genotypes probabilities output file (see Section 2.2), except that the sample identifiers are listed only once (one column per sample). The remaining lines give data for each marker (one marker per line). The first three columns of the genotype dosage file are identical to the first 3 columns in the genotype probabilities output file (see Section 2.2). The remaining columns give the estimated B-allele dosages for each marker and each sample (one sample per column).

5.11 Phased genotypes files [.phased.gz]

Output phased haplotypes are produced by BEAGLECALL when the runbeagle parameter is true. A GZIP-compressed phased genotypes file is created for each cohort specified with a signals argument in the BEAGLECALL command line (Section 4.1). The phased genotypes files give estimated most likely haplotypes for each sample. The estimated haplotypes can be used in downstream analysis. The output phased haplotypes are in BEAGLE format with rows corresponding to markers and columns corresponding to haplotypes. See the documentation to the BEAGLE software package for more details.

5.12 Estimated allelic R^2 files [.r2]

When the runbeagle parameter is true, an estimated allelic R^2 file is created for each cohort specified with a signals argument in the BEAGLECALL command line (Section 4.1). The estimated allelic R^2 file reports the estimate squared correlation between the most likely genotype and the true genotype for each marker. See the BEAGLE documentation (http://www.stat.auckland.ac.nz/~bbrowning/beagle/beagle.html) for more details.[5]

6 Acknowledgements

BEAGLECALL incorporates software from the Apache Commons Mathematics Library (http://commons.apache.org/math/). The Apache Commons Mathematics Library is licensed under the Apache License, Version 2.0 (http://www.apache.org/licenses/LICENSE-2.0).

BEAGLECALL incorporates a Java port of Jan Wigginton's C/C++ SNP-HWE[4] function that is available from http://www.sph.umich.edu/csg/abecasis/Exact/. The algorithm for the exact test of Hardy-Weinberg equilibrium implemented in SNP-HWE was developed by J. E. Wigginton, D. J. Cutler, and G. R. Abecasis.[4]

We thank Y. Y. Teo and T. G. Clark for their permission to use the example allele signal intensity data that is included in the ILLUMINUS software distribution.[1]

7 References

1. Teo, Y.Y., Inouye, M., Small, K.S., Gwilliam, R., Deloukas, P., Kwiatkowski, D.P., and Clark, T.G. (2007). A genotype calling algorithm for the Illumina BeadArray platform. Bioinformatics *23*, 2741-2746.
2. Rabbee, N., and Speed, T.P. (2006). A genotype calling algorithm for affymetrix SNP arrays. Bioinformatics *22*, 7-12.
3. The Wellcome Trust Case Control Consortium (2007). Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. Nature *447*, 661-678.
4. Wigginton, J.E., Cutler, D.J., and Abecasis, G.R. (2005). A note on exact tests of Hardy-Weinberg equilibrium. Am J Hum Genet *76*, 887-893.
5. Browning, B.L., and Browning, S.R. (2009). A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals. Am J Hum Genet *84*, 210-223.
6. Browning, S.R., and Browning, B.L. (2007). Rapid and accurate haplotype phasing and missing-data inference for whole-genome association studies by use of localized haplotype clustering. Am J Hum Genet *81*, 1084-1097.