IBDseq

Brian L. Browning
Department of Medicine
Division of Medical Genetics
University of Washington

November 7, 2013

1 Introduction

IBDseq is a software program for detecting segments of identity-by-descent (IBD) and homozygosity-by-descent (HBD) in unphased genetic sequence data. IBDseq can analyze large data sets with thousands of individuals, and it can analyze data with multi-allelic markers.

IBDseq requires a Java 1.6 interpreter (or a later version). A Java interpreter is probably already installed on your computer (type "java –version" at the command line prompt to check). The Java interpreter can be downloaded from the **java.sun.com** web site.

1.1 Citing IBDseq

If you use IBDseq and publish your analysis, please report the version of the program used and please cite the following publication:

B L Browning and S R Browning (2013) Detecting identity by descent and estimating genotype error rates in sequence data. The American Journal of Human Genetics 93(5):840-851. http://dx.doi.org/10.1016/j.ajhg.2013.09.014

1.2 Files in the IBDseq software distribution

The IBDseq software package is open-source and freely available and can be downloaded from the IBDseq web site:

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

1.3 Variant Call Format

Beagle uses Variant Call Format (VCF) 4.1 for input and output file. VCF files can be manipulated and analysed with VCFtools, PLINK/SEQ, and the Beagle Utilities.

VCF input files can be compressed with the GZIP algorithm. IBDseq assumes that any file that has a name ending in ".gz" is compressed with GZIP.

2 Command line arguments

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

```
java –Xmx[Mb]m -jar ibdseq.jar [arguments]
```

where [Mb] is the number of megabytes of memory allocated for the analysis (e.g. −Xmx2000m) and [arguments] is a space separated list of arguments. Each argument has the format **parameter=value**. There is no white-space between the **parameter** and **=** or between **=** and the **value**. Large data sets with thousands of samples may require several gigabytes of memory.

IBDseq has two required arguments: the **gt** argument to specify an input VCF file, and the **out** argument to specify the output file prefix. Other command line arguments are optional and have reasonable default values. The **nthreads** argument enables parallel computation.

2.1 Arguments for specifying input data

- **gt=**[file] specifies a VCF file containing a GT format field for each marker. The **gt** argument is required. Phase information in the VCF file is ignored.
- **out=**[prefix] specifies the output filename prefix. The prefix can be an absolute or relative filename, but it cannot be a folder name. The **out** argument is required.
- **excludesamples=**[file] specifies a file containing sample identifiers (one identifier per line) that will be excluded from the analysis and output files.
- **excludemarkers=**[file] specifies a file containing marker identifiers (one identifier per line) that will be excluded from the analysis and output files. The marker identifier in the exclude markers file can either be an identifier from the VCF record's ID field, or CHROM:POS where CHROM and POS are the VCF record's CHROM and POS fields.
- **chrom=**[chrom:start-end] specifies the starting and ending coordinates of a chromosome interval to be analysed. The chromosome identifier must match the chromosome identifier used in the VCF file. The entire chromosome, the beginning of the chromosome, and the end of a chromosome can be specified by **chrom=**[chrom], **chrom=**[chrom:-end], and **chrom=**[chrom:start-] respectively. If a **chrom** argument is not specified, the first chromosome in the VCF file will be analysed.

2.2 Other arguments

- **nthreads=**[positive integer] specifies the number of threads of execution to use in the analysis. On multi-core machines, increasing the value of the **nthreads** parameter (up to the maximum available cores) can reduce computation run time (default: **nthreads=1**).
- **ibdlod=**[positive number] specifies minimum LOD score for IBD segment printed to the output IBD file (default: **ibdlod=3.0**).
- **ibdtrim=**[nonnegative number] controls the trimming of the ends of the IBD segment. At each end of a detected IBD segment, the trim will be the largest possible trim in which the trimmed markers contribute a LOD score less than the **ibdtrim** parameter. The IBD segment reported to the output IBD file will have the before-trimming LOD score and the after-trimmed end points. (default: **ibdtrim=0.0**).
- **errormax=**[non-negative number] specifies the maximum allele error rate (default: **errormax=0.001**) in the analysis. The analysis allele error rate for a marker is the minimum of the **errormax** and the product of the **errorprop** parameter and the minor allele frequency.
- **errorprop=**[non-negative number] specifies the maximum ratio of the analysis allele error rate and the minor allele frequency (default: **errorprop=0.25**). The analysis allele error rate for a marker is the minimum of the **errormax** parameter and the product of the **errorprop** parameter and the minor allele frequency.
- **r2window=**[non-negative integer] specifies the number of markers in the sliding window used for detecting correlated markers (default: **r2window=500**).
- **r2max=**[number between 0 and 1 inclusive] specifies the maximum permitted squared correlation of minor allele dosage. If two markers have squared correlation greater than

the specified **r2max** parameter, the marker with the higher minor allele frequency will be excluded from the analysis (default: **r2max=0.15**).

- **minalleles=**[integer ≥ 2] specifies the minimum number of samples carrying the minor allele. If a marker has fewer than the minimum number of minor allele carriers, the marker will be excluded from the analysis (default: **minalleles=2**). For multi-allelic markers, the minor allele is the allele with the second largest allele frequency.

3 Output files

Three output files are created whose names begin with the prefix specified in the **out=** command line argument and end with a descriptive suffix (".log", ".ibd", ".hbd", or ".r2.filtered").

3.1 log file [.log]

The log file contains a summary of the IBDseq analysis. The log file reports the number of samples and the number of markers before and after filtering. A marker is excluded if it has too few minor allele carriers (see the **minalleles** argument) or if it is correlated with a non-excluded marker. The log file also reports the mean number of IBD segments per sample, the mean IBD segment length, and the mean number of IBD segments covering a genotype (the "IBD depth").

3.2 R2-filtered file [.r2.filtered]

The R2-filtered file lists markers (one marker per line) that were excluded from contributing to the IBD segment LOD scores because of inter-marker correlation. The R2-filtered file contains the first five fields of the VCF records for the excluded markers.

3.3 IBD file [.ibd] and HBD file [.hbd]

Each line of the IBD output file represents an IBD segment. Each line of the HBD output file represents an HBD segment. Each line of an HBD or IBD output file has 8 tab-delimited fields:

1) First sample identifier
2) First sample haplotype index (0, 1, or 2; 0 = unknown)
3) Second sample identifier
4) Second sample haplotype index (0, 1, or 2; 0 = unknown)
5) Chromosome
6) Starting genomic position (inclusive).
7) Ending genomic position (inclusive).
8) LOD score (larger values indicate greater evidence for the IBD segment).

The two haplotype indices will be 0 for IBD segments and will be 1 and 2 for HBD segments.

4 References

1. Danecek, P., et al., *The variant call format and VCFtools.* Bioinformatics, 2011. **27**(15): p. 2156-2158.