Manual for mcmcPed
Software for Estimating Pedigrees and Short-Term Effective Population Size Using SNP or Microsatellite data.

Amy Ko
amyko@berkeley.edu

December 4, 2018

Contents

1 PLEASE NOTE!

mcmcPed is designed to estimate outbred pedigrees and therefore will not necessarily work well for highly inbred pedigrees. We suggest you first check the level of inbreeding in your samples before proceeding with pedigree estimation.

Furthermore, likelihood computation implemented in mcmcPed is sensitive to linkage disequilibrium (LD) so we suggest you prune the markers before using our software. For human data, we found 10,000 to 20,000 SNPs to work well. Alternatively, you can provide likelihood values directly as input (See 4.2 for details).

Future releases of the program will aim to account for inbreeding and LD.

2 Description

mcmcPed jointly estimates the pedigree of a sample of individuals and the short-term effective population size (N_e) from SNP or microsatellite data. MCMC is used to estimate the joint posterior probability of the pedigree and N_e. We assume that individuals are outbred and that there are no complex cyclic relationships such as double first cousins. To estimate population allele frequencies, the method requires several individuals to be present in the data. Alternatively, allele frequenceis can be provided directly as input. The method uses only autosomal diploid chromosomes and so any sex chromosomes must be removed from the input files.

3 Installation

The latest versions of our software and manual can be downloaded at `https://github.com/amyko/mcmcPed`. The download should include two folders: data and src. Go into the source folder (src) and make sure that the script "mcmcPed" is executable:

```
cd mcmcPed-master/src
chmod +x mcmcPed
```

4 Usage

4.1 Input Files

The program takes PLINK-formatted TPED and TFAM files as input.

4.1.1 yourFileName.tped

The alleles in the TPED file can be represented by any characters, except "0" which is reserved for missing data. The TPED is a white-space delimited file that contains the following columns: 1) chromosome, 2) marker, 3) genetic distance (Morgan), 4) physical distance (bp), and genotypes of each individual. The TPED file does not contain a header. For example, the following TPED file contains the genotypes of 2 individuals at 3 markers.

```
1   snp1   0.026   2014219   1 1 2 1
1   snp2   0.032   2449448   1 2 2 2
1   snp3   0.047   3652230   2 1 2 1
```

N.B. All markers should be polymorphic in the reference population even if they may not be polymorphic in the samples.

4.1.2 yourFileName.tfam

The TFAM file contains 1) family ID, 2) individual ID, 3) paternal ID, 4) maternal ID, and 5) sex (1=male, 2=female). Missing paternal or maternal ID is encoded as "0". Here's an example of a TFAM file for 3 individuals:

```
fam1   indiv1   0   0   1
fam1   indiv2   0   0   2
fam1   indiv3   0   0   1
```

4.1.3 yourFileName.freq (OPTIONAL)

In addition, you can provide a optional frequency file that contains the allele frequency of each marker in the TPED file. If the frequency file is not provided, then the allele frequencies will be estimated from the TPED file.

The frequnecy file contains $2m$ lines, where m is the number of markers represented in the TPED file. The first line is the list of alleles for the first marker and the second line is the corresponding allele frequencies, and so on. Here's an example of a frequency file for 3 markers.

```
 1    2
.2   .8
 1    2    3
.1  .05  .85
 1    2
.7   .3
```

The first marker has two alleles (1 and 2) and has corresponding allele frequencies of .2 and .8; the second marker has three alleles (1,2, and 3) and frequencies of .1, .05, and .85, and so on. Note that for each marker, the frequencies sum

to 1.

mcmcPed will then use the TPED and TFAM files to compute the marginal and pairwise likelihoods for the samples, which in turn will be used in MCMC to estimate the pedigree.

4.2 Alternative Input Files (.marginal and .pairwise)

Alternatively, you can provide files containing marginal and pairwise likelihoods directly as input using pairwise likelihood computation method of your choice (e.g. [2, 3]). If you choose this option, you still need to provide the TFAM file described in 4.1.2, along with marginal (*.marginal*) and pairwise (*.pairwise*) likelihood files described below.

4.2.1 yourFileName.tfam

See 4.1.2 for format.

4.2.2 yourFileName.marginal

The marginal likelihood file contains the log marginal likelihood of each sample per line. The order of the likelihood values must match the order of the individuals in the TFAM file. Therefore, the number of lines in this file should be equal to the number of individuals in the sample. For example, the marginal likelihood file for three individuals would look something like this:

-1762.60
-1784.04
-1782.60

where the first line corresponds to the first individual in the TFAM file, the second line corresponds to the second individual, and so on.

4.2.3 yourFileName.pairwise

The pairwise likelihood file contains the log pairwise likelihood values for all pairs of individuals for each possible relationship type. The relationship type is denoted by " > ", followed by three integers: 1) number of generations between individual 1 and the common ancestor, 2) number of generations between individual 2 and the common ancestor, and 3) the number of common ancestors between individual 1 and 2. For example, an avuncular relationship is denoted by "> 2 1 2." The table below shows the relationship keys for all pairwise relationships that span up to 5 generations.

Relationship	Key
Unrelated	0 0 0
Full siblings	1 1 2
Half siblings	1 1 1
Full cousins	2 2 2
Half cousins	2 2 1

Following the keys shown this table, the pairwise file must contain *all possible pairwise relationships for the maximum number of generations you specify* in the option file (See Section 5). For example, if the maximum generation you specify is 2, then the pairwise file must contain values corresponding to unrelated, parent-offspring, full siblings, and half siblings.

After you specify the relationship type, the subsequent lines correspond to the log likelihood values for $n(n-1)/2$ pairs of samples. The first two columns of the file are the indices of the two individuals, where the index $0, 1, 2, ...$ correspond to $indiv1, indiv2, indiv3, ...$, respectively, specified in the TFAM file. For example, the following shows the pairwise likelihood values for "unrelated" and "full-sibling" relationships for 3 individuals.

```
> 0   0   0
0     1   -3546.65
0     2   -3545.20
1     2   -3487.71
> 1   1   2
0     1   -3332.10
0     2   -3300.49
1     2   -3658.31
```

4.3 Run Command

We can run the program by providing an option file which contains the name of the input files and other various parameters (See Section 5 for the format of the option file).

```
./mcmcPed myOptionFile
```

4.4 Run example

Here we describe how to run the program on the provided example data. The example data is located in mcmcPed-master/data.

```
cd data
```

The data folder contains test files containing 50 individuals (test.tped and test.tfam) and an option file that specifies various parameters for the program

(options). From the data folder, we can run the program by calling "mcmcPed":

```
../src/mcmcPed mcmcOptions
```

This creates five output files, which are discussed in Section 6.

5 Options

This section describes various options you can include in the option file. Each line in the option file corresponds to a particular option described below (the order of parameters does not matter). The first column is the value of the option and the second column is the option name. The option name is always preceded by "#" (e.g. #fileName). The first and second columns are separated by a white space delimiter.

#fileName Name of TPED and TFAM files. For example, if the files are named "myFile.tped" and "myFile.tfam", then the value for the fileName option should be "myFile". N.B. The full path has to be provided.

#outFileName Name of output file name. N.B. The full path has to be provided.

#computeLikelihood Indicator (0 or 1) denoting whether to compute marginal and pairwise likelihoods internally. If it is set to 0, then the the user must provide *.marginal* and *.pairwise* files (See 4.2 for format of these files). If set to 1, the likelihood files are generated from the genotype data given by *.tped* files. (default = 1)

#indep Indicator (0 or 1) denoting whether the markers are independent. If they are independent (indep=1), then method by [4] is used to compute the likelihoods. If markers are not independent (indep=0), then the HMM by [2] is used. Furthermore, the genetic positions must be provided in the *.tped* file if indep is set to 0.

#epsilon1 Allele drop out rate. (default=0)

#epsilon2 Sequencing error rate. (default=0.01)

#maxGen Maximum number of generations spanned by the pedigree. For example, to infer up to first cousins, maxGen should be set to 3. For sibship inference only, maxGens should be set to 2. Our method supports up to 3 generations. (default=3)

#minN Lower bound for the population size that MCMC explores (default=5).

#maxN Upper bound for the population size that MCMC explores (default=5000).

#sdN Standard deviation for the proposal distribution for the population size (N). That is, the next value of N is drawn from the normal distribution $N(currentN, sdN)$

#minAlpha Lower bound for alpha that MCMC explores (default=.1).

#maxAlpha Upper bound for alpha that MCMC explores (default=20).

#sdAlpha Standard deviation for the proposal distribution for alpha. That is, the next value of alpha is drawn from the normal distribution $N(currentAlpha, sdAlpha)$. (default=2)

#minBeta Lower bound for beta that MCMC explores (default=.00001).

#maxBeta Upper bound for beta that MCMC explores (default=.1).

#sdBeta Standard deviation for the proposal distribution for beta. That is, the next value of beta is drawn from the normal distribution $N(currentBeta, sdBeta)$. (default=.01)

#burnIn Number of burn-in samples (default=4000000).

#numSamples Number of MCMC samples (default = 2000000).

#sampleFreq Sample frequemcy for the MCMC samples. For example sample frequency of 50 means we save every 50th MCMC sample. (default = 50)

#numRuns Number of independent MCMC runs. (default=1)

#numThreads Number of threads to use. (default=1)

6 Output files

There are five output files for each run of the program: *.pedigrees, .pedCount, .theta, .Ne,* and *.pairAssignment*. For example, for a output file named myFile and run number 2, the output files will be *myFile.2.pedigrees, myFile.2.pedCount*, and so on.

6.1 pedigrees

This file contains pedigrees sampled by MCMC. The first line begins with a special character ">", followed by the ID number and the log posterior probability of the pedigree described in the subsequent lines. For example, the first line with pedigree ID number of 0 and log posterior probability value of -352585 would look like:

> 0 -352585

In the subsequent lines, there are five columns:

Name Name of the individual. For sampled individuals, the name is "FID_IID" from the input file (e.g. familyA_indiv6). Unsampled individuals are named "unsampled_x," where x is the individual ID.

FATHER Name of the father. 0 means missing.

MOTHER Name of the mother. 0 means missing.

SEX Sex of the individual (M = male, F = female).

SAMPLED Sample status of the individual (1 = sampled, 0 = not sampled).

After these lines, the next pedigree is denoted by "> 1 loglikelihood", and so on. Here's an example of what the *.pedigrees* file would look like for two pedigrees of two sampled individuals:

>	0	-352585		
NAME	FATHER	MOTHER	SEX	SAMPLED
1_0	unsampled_0	unsampled_1	M	1
1_1	unsampled_0	unsampled_1	F	1
unsampled_0	0	0	M	0
unsampled_1	0	0	F	0
>	1	-352579		
NAME	FATHER	MOTHER	SEX	SAMPLED
1_0	0	0	M	1
1_1	0	0	F	1

Here, the two sampled individuals (1_0 and 1_1) are full siblings in pedigree 0; they are unrelated in pedigree 1.

6.2 pedCount

This file contains the number of times each pedigree was sampled during the MCMC run. The file has two columns:

pedID ID of pedigree. This ID corresponds to the pedigree ID in the *.pedigrees* file.

count Number of times the pedigree was sampled.

6.3 theta

This file contains the sampled mating parameters. Each line corresponds to a single sample. The file contains four columns:

N Population size.

alpha alpha.

beta beta.

Ne Effective population size computed from N, alpha, and beta.

6.4 pairAssignment

This file contains information about the pairwise relationship for each pair of individuals in the sample. The file has 9 columns:

FID1 FID of individual 1.

IID1 IID of individual 1.

FID2 FID of individual 2.

IID2 IID of individual 2.

nFS Number of times the two individuals were full siblings in the MCMC samples.

nHS Number of times the two individuals were half siblings in the MCMC samples.

nUR Number of times the two individuals were unrelated in the MCMC samples.

nFC Number of times the two individuals were full first cousins in the MCMC samples.

nHC Number of times the two individuals were half first cousins in the MCMC samples.

6.5 lkhd

This file contains the log likelihood values over the course of the MCMC run. The file has two columns, where the first column is the iteration number and the second column is the composite likelihood of the pedigree at that iteration. Note that the first iteration number in this file is equal to burnIn since sampling begins at the end of the burn-in period.

nIter Number of iterations

logLikelihood Log posterior probability

We recommend that you examine the likelihood values from multiple runs to check whether the log likelihood values from all the runs fluctuate in a similar range. If not, consider increasing the burn-in parameter described in Section 5.

6.6 Visualizing the data

The estimated pedigree encoded in *.pedigrees* file can be drawn by pedigree drawing softwares such as FamAgg [5]. Below is an example of how we could draw a pedigree in R using FamAgg:

```
library("FamAgg")

data <- read.table("myPedigreeFile", header = TRUE)
fad <- FAData(pedigree = data)
plotPed(fad, family='1')
```

where "myPedigreeFile" is a text file containing the information about the pedigree of interest:

family	id	father	mother	sex
1	1	3	4	M
1	2	3	5	F
1	3	0	0	M
1	4	0	0	F
1	5	0	0	F

Figure 1 shows the resulting pedigree diagram:

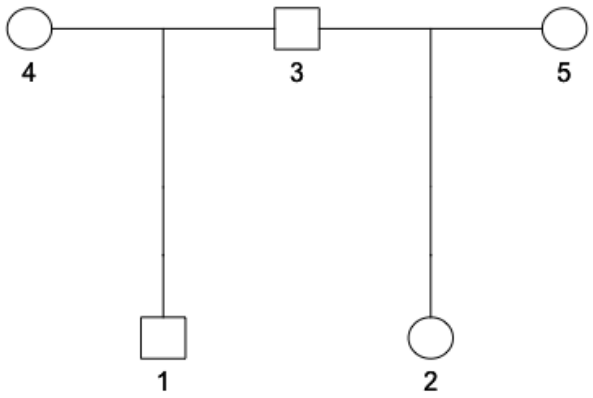

Figure 1: Pedigree drawn by FamAgg, where the input data was given by "myPedigreeFile" (see Section 6.6).

7 Bias Correction for N_e

If the sample contains relatives beyond first cousins, it is possible that N_e estimation may be biased. One appraoch for correcting the bias is through simulations. Let $S_{ibd} = nFS + .5 * nHS$ denote the level of overall IBD sharing computed from the siblings inferred by the method. Here, nFS and nHS are the number of full sibling pairs and half siblings pairs, respectively, estimated by the method. We then simulate pedigrees under various values of N_e and compute the corresponding S_{IBD} for each N_e. We then seek N_e whose S_{IBD} most closely matches the S_{IBD} from the real data.

Here's an illustration of how to run simulations to correct the potential bias in N_e. Run the following command from the example data folder:

```
../src/simulatePedigrees simOptions
```

where *simOptions* is a file containing various parameters for the simulation:

#fileName Name of *.pairAssignemnt*, the output file from running MCMC. N.B. The full path has to be provided.

#outfileName Name of outfile name, where the simulation results will be saved. N.B. The full path has to be provided.

#alpha Alpha value under which pedigrees will be simulated. For example, you can use the mode value of alpha from *.theta* file from MCMC.

#beta Beta value under which pedigrees will be simulated. For example, you can use the mode value of beta from *.theta* file from MCMC.

#sampleSize Number of individuals in the sample. (i.e. number of individuals in the TFAM file)

#maxN Maximum value for population size (N) under which pedigrees will be simulated. (default=10000)

An example output file is shown below:

sibd computed from myData.pairAssignment:
5.0

Simulation results:

Ne	mean_sibd	se_sibd
257	9.910	2.984
289	8.110	2.722
322	7.550	2.519
354	6.680	2.101
387	6.170	2.542
419	6.040	2.009
452	5.820	2.765
485	4.910	2.253
517	4.680	2.009
550	4.380	2.300
583	4.400	2.012

Here, S_{IBD} computed from the real data was 5.0 as shown in the first two lines. The subsequent lines show the results from simulations. The three columns for the simulation results are: N_e, average S_{IBD}, and the standard error. From the simulations, we see that $N_e = 485$ has average S_{IBD} that most closely matches S_{IBD} from the real data.

8 Recommendations for preprocessing the data

The current version of our program does not support missing genotypes. When missing data is encountered during the likelihood computation, the marker is simply skipped. Incorporation of missing data will be implemented in the next version of the software.

Furthermore, we recommend that you prune markers to reduce the effects of LD to prevent overestimation of relatedness. One way to do this is using PLINK [1].

```
plink –tfile myInputFile –indep-pairwise 50 5 .05
```

The above command prunes the set of markers in myIntputFile.tped at $r^2 = .05$ with a window size of 50 SNPs and step size of 5. It outputs "plink.prune.in" which contains the filtered markers. Appropriate values for r^2 and the window size depend on the genome length. For human genomes, our simulations showed that $r^2 = .05$ and window size equivalent of .04 cM work well.

Now we can extract the markers contained in "plink.prune.in" from the original set of markers:

```
plink –tfile myInputFile –extract plink.prune.in –make-bed –recode –tab
–transpose –out myInputFile.LDpruned
```

which creates myInputFile.LDpruned.tped and myInputFile.LDpruned.tfam files, where LD markers are pruned away.

References

[1] Chang CC, Chow CC, Tellier LC, Vattikuti S, Purcell SM, Lee JJ. Second-generation PLINK: rising to the challenge of larger and richer datasets. Gigascience. 2015;4:7. doi:10.1186/s13742-015-0047-8.

[2] Albrechtsen A, Sand Korneliussen T, Moltke I, van Overseem Hansen T, Nielsen FC, Nielsen R. Relatedness mapping and tracts of relatedness for genome?wide data in the presence of linkage disequilibrium. Genetic epidemiology. 2009 Apr 1;33(3):266-74.

[3] Abecasis GR, Cherny SS, Cookson WO, Cardon LR. Merlin rapid analysis of dense genetic maps using sparse gene flow trees. Nature genetics. 2002 Jan 1;30(1):97-101.

[4] Weir BS, Anderson AD, Hepler AB. Genetic relatedness analysis: modern data and new challenges. Nature Reviews Genetics. 2006 Oct;7(10):771.

[5] Rainer J, Taliun D, D'elia Y, Pattaro C, Domingues FS, Weichenberger CX. FamAgg: an R package to evaluate familial aggregation of traits in large pedigrees. Bioinformatics. 2016 Jan 22;32(10):1583-5.