Rolf Nevanlinna Institute

BAMA Reference Manual

Bayesian Analysis of Multilocus Association / Quantitative and Qualitative Traits

---

Initial version May 26, 2009

This software is a computer implementation of the Bayesian multilocus association method of Kilpikari and Sillanpää (2003). The program implements the Metropolis-Hastings-Green (Metropolis et al. 1953, Hastings 1970, Green 1995) algorithm in estimation of the model parameters (see the above paper for further detail). The software is written in C-language and is designed for Unix or Linux environment but may work also in other environments.

Please address all the correspondence to:

Mikko J. Sillanpää
Rolf Nevanlinna Institute,
Department of Mathematics and Statistics,
P.O. Box 68, FIN-00014 University of Helsinki, Finland

email : `mjs@rolf.helsinki.fi` web : http://www.rni.helsinki.fi/∼mjs

Technical details

The main file named `bama.c` includes following four lines for Sun UltraSparc workstations:

```
long lrand48(void);
#define rand() lrand48()
#undef RAND_MAX
#define RAND_MAX 2147483647
```

These lines should be commented out when the software is compiled in some Linux environments or in such Unix systems which have different RAND_MAX values. If these lines are omitted in Sun UltraSparc workstation or used in Linux, the software may not work properly and can give wrong results. Before compilation, please check what is a proper RAND_MAX value in your system.

Compiling the program

You can compile the program with `gcc` compiler with the command:

```
> gcc -o bama bama.c -Wall -pedantic -lm
```

Additionally, by using option `-O` here will optimize and speed up the execution of the BAMA program in practice. Because the program do not use dynamic memory allocation, user must specify first large enough sizes for the constants in the start of the program bama.c. If the program do not compile properly, by linux command `> limit` one can check allocation for stacksize. By command

```
> limit stacksize 81920
```

one can extend size of the stack to be larger (e.g., 81920) and try to compile the program again.

Registration & Mailing list

No official registration for the execution of the program is required, however, one may inform me that one is willing to joint for the mailing list. It would be appreciate if one would sent me an e-mail message subjected as 'participation to BAMA mailing list' where she/he would

indicate his/her name, institution and e-mail address. I will update given names to the mailing list. This way I could notify the people in the list about a newer version of the program and possible errors in the program. (One may find the same information from my web-page as well.) This information would also give me some feedback how many persons are using the program or are interested in it.

BAMA input files

Six input files are required in your working directory before you can start running the program. These files are :

(1) **N_markers.bap**
(2) **all_markers.bap**
(3) **asm_markers.bap**
(4) **index.bap**
(5) **phenotypes.bap**
(6) **directives.bap**

Contents and formats of the files are presented in the following sections.

Number of markers and chromosomes - (N_markers.bap)

`Current version of the program works only if you specify that all your markers are in the first chromosome!!!!`

Total number of chromosomes and numbers specifying how many markers you have on each chromosome (in your data) has to be specified in file `N_markers.bap`. This file includes separate lines for each given number. In the first row, number of separate chromosomes in your data is given. Subsequent rows contain information on number of markers per chromosomes.

An example file of 5 chromosomes is given below. There are 15 markers in chromosome 2 and 10 markers in all the other chromosomes.

```
12345678901234567890123456789 <- column number
5          < - file starts from this line
10
15
```

```
10
10
10          < - file ends at this line
```

Genotype data of the markers - (all_markers.bap)

The input file for marker data has to be named as `all_markers.bap`. The program assumes that all genotypic information (on all markers) is given at this single file. This file can be created by concatenating separate marker files vertically together by using unix command `cat`. Each row (corresponding to single individual) in the file should contain following information separated with space: family id, individual id, allele 1 in marker 1, allele 2 in marker 1, allele 1 in marker 2, allele 2 in marker 2, and so on. Individual id must be some positive integer. It is required that the alleles are coded as consecutive numbers and their numbering starts from one. This program is cabable of handling missing values in genotype data. The code used for missing allele is -1. Note that the number of segregating alleles specified in file `asm_markers.bap` has to be in agreement with the information given in this file. (Family id is redundant information but is required here. You can systematically use number 1 here.)
An example file of 5 individuals and 6 markers is given below.

```
12345678901234567890123456789 <- column number
1 1 1 1 1 2 3 3 1 1 2 3 2 2              < - file starts from this line
1 2 1 1 2 2 4 3 5 1 3 3 2 2
1 3 1 1 3 3 4 4 1 5 2 2 5 1
1 4 1 1 1 2 3 3 4 3 4 4 1 1
1 5 1 1 3 2 3 3 1 5 2 4 1 2             < - file ends at this line
```

Dominant markers or coding of two-locus haplotypes (CF-example in Kilpikari and Sillanpää 2003) may result in partial (genotypic or) allelic information. If only one allele can be observed and other allele can be any allele, then use simply code -1 for unobserved allele. Other kind of partial allelic information can be utilized here as follows. The two-digit missing allele code -23 is interpretated so that missing allele can equally be either 2 or 3. Similarly code -14 means that missing allele can be either 1 or 4. Other codes such as -13 or -24 are also possible (see CF-example). The program utilizes this information in data augmentation. If no partial information

about missing allele is available, use code -1 for missing allele.

Analyzing only subset of given markers - (asm_markers.bap)

This file determines the subgroup of markers which are included in your association analysis from the file `all_markers.bap`. You can of course include all the markers in your analysis if you like. Additionally the information on how many segregating alleles each marker has is given here.

The number of candidate markers to be analyzed is specified in the first line of this file. Then, each following line contain information on: a chromosome number, a marker number, and a number specifying how many alleles is segregating in this marker. Note that numbering of chromosomes and markers must start from one.

An example file of 5 markers in chromosome 1 is given below. There are 3,4,8,6, and 5 alleles in these 5 markers.

```
12345678901234567890123456789 <- column number
5              < - file starts from this line
1 1 3
1 2 4
1 3 8
1 4 6
1 5 5              < - file ends at this line
```

Analyzing only subset of given individuals - (index.bap)

This file determines the subgroup of individuals which are included in your association analysis from the marker (`all_markers.bap`) and phenotype (`phenotypes.bap`) files. You can of course include all the individuals in your analysis if you like.

First row in the file contains information on the number of selected individuals used in analysis and number of all individuals in the data. Then each following line contain information on indi-

vidual id. Note that individual id (i.e. identification number) must correspond to one that was used in file `all_markers.bap`.

An `index.bap` example file of selecting 5 individuals (out of total 5) is given below.

```
12345678901234567890123456789 <- column number
5 5              < - file starts from this line
1
2
3
4
5              < - file ends at this line
```

Phenotype data - (phenotypes.bap)

Phenotypes are read in from file `phenotypes.bap`. Each individual has own line in this file. Each row (corresponding to single individual) in the file should contain following information separated with space: family id, individual id, trait1, trait2, trait3, .. Family id and individual id should correspond to numbers used in file `all_markers.bap`. Number of different traits here must correspond to given number of traits in `directives.bap`. This file contains complete set of phenotypes.

An `phenotypes.bap` example file with 5 individuals and one quantitative trait:

```
12345678901234567890123456789 <- column number
1  1   4.537248             < - file starts from this line
1  2   4.411541
1  3   3.135545
1  4   6.576165
1  5   4.023658             < - file ends at this line
```

An `phenotypes.bap` example file with 5 individuals and one binary trait:

```
12345678901234567890123456789 <- column number
1  1   0.00             < - file starts from this line
1  2   1.00
```

```
1  3   1.00
1  4   0.00
1  5   1.00              < - file ends at this line
```

Directives for MCMC estimation - (directives.bap)

To control MCMC estimation, user defines here the number of MCMC rounds, thinning, type of trait to be analyzed an so on.

The directives.bap has a following format:

1. The number of MCMC rounds to be run.
2. Thinning (x) of the chain. Only every x^{th} iteration is printed out to the file.
3. Type of trait to be analyzed { 1= quantitative trait
   2= binary trait (e.g., disease status) }
4. Number of traits in the complete data set.
5. A trait number (controls which trait is analyzed). Note that numbering must start from one.
6. Lower and upper bounds for the uniform prior of regression coefficients (allelic effects).
7. Lower and upper bounds for the uniform prior of residual variance (σ^2). For a binary trait, you can place any two values here because $\sigma^2 = 1$ (constant).
8. A proposal range of regression coefficients (allelic effects).
9. A proposal range of residual variance (σ^2). For binary traits, you can place any value here.
10. This line is not used. Use 0 here.

Note that a natural lower bound for σ^2 is zero and upper bound is the phenotypic variance.

An directives.bap example file is given below.

```
12345678901234567890123456789 <- column number
300000              < - file starts from this line
10
1
1
```

```
1
-100.0  100.0
0.0     3.0
0.8
0.5
0               < - file ends at this line
```

Program outputs

1) Rejection rates and input parameter values are collected into a file named `rejection_rates.txt`.

2) In each MCMC round, the program prints out a value of the variable N_{qtl} (number of QTLs) into a ASCII file named `N_qtl.txt`.

3) The program prints out QTL locations from each MCMC round into file `mcmc_output.txt`. If the number of QTLs in the current MCMC round is zero, this file is not updated, if it is one, only one line (marker number) is printed into file, if it is two, two lines (marker numbers) are printed into file and so on. Each occupied marker position (QTL) is printed as separate line into this file. Numbering of markers among candidate loci starts from one.

4) The program prints out a value of residual variance (σ^2) into file named `a5.txt`.

Matlab

Visualisation of and construction of the QTL-probabilities from MCMC output files, can easily be done with Matlab software. Some example scripts that can be run under Matlab are found below.

Plot of sample path for the number of QTLs:

```
load N_qtl.txt

plot(N_qtl.txt)
```

Plot of posterior distribution for different numbers of QTLs:

```
load N_qtl.txt
[n]=hist(N_qtl,6)
n=n/size(N_qtl,1)
bar(n)
```

To avoid missinterpretations, one should first check that each possible value is represented in the distribution. Note that the prior distribution assumed for the number of QTLs in the method (and the software) is actually an accelerated truncated Poisson distribution instead of ordinary Poisson distribution (see Sillanpää et al. 2004).

plotting posterior QTL-probabilities :

```
n_markers=5
n_mcmc=100 000
load mcmc_output.txt
[n]=hist(mcmc_output,n_markers)
n=n/n_mcmc
bar(n)
```

Here `n_markers` is the number of candidate markers and `n_mcmc` is the number of MCMC rounds (after thinning) in your analysis.

Dominance, epistasis, related individuals and more

It is possible to use this software to search for pairwise epistatic interactions and allowing dominance effects in your model. This can be done by coding your input data in the certain way. For details of this, see Sillanpää (2009). It is also good to know that the Bayesian multiple QTL model used in the software should be relatively robust to the use of related individuals in your

data (see Pikkuhookana and Sillanpää 2009). If your data contains highly dependent marker set or genetic heterogeneity may be the problem, it may be good idea to do analyses proposed by Sillanpää and Bhattacharjee (2005,2006) using WinBUGS software.

References

Green, P. J. (1995) Reversible jump Markov Chain Monte Carlo computation and Bayesian model determination. Biometrika **82**: 711-732.

Hastings, W. K. (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika **57**: 97-109.

Kilpikari, R. and M. J. Sillanpää (2003) Bayesian analysis of multilocus association in quantitative and qualitative traits. Genetic Epidemiology **25**: 122-135.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (1953) Equation of state calculations by fast computing machines. Journal of Chemical Physics **21**: 1087-1092.

Pikkuhookana, P., and M. J. Sillanpää (2009) Correcting for relatedness in Bayesian models for genomic data association analysis. Heredity (accepted).

Sillanpää, M. J. (2009) Detecting interactions in association studies by using simple allele recoding. Human Heredity **67**: 69-75.

Sillanpää, M. J. and M. Bhattacharjee (2005) Bayesian association-based fine mapping in small chromosomal segments. Genetics 169: 427-439.

Sillanpää, M. J. and M. Bhattacharjee (2006) Association mapping of complex trait loci with context-dependent effects and unknown context-variable. Genetics 174: 1597-1611.

Sillanpää, M. J., D. Gasbarra, and E. Arjas (2004) Comment on the "On the Metropolis-Hastings acceptance probability to add or drop a quantitative trait locus in Markov chain Monte Carlo-based Bayesian analyses". Genetics **167**: 1037-1037.