
EMIBD9 Software Manual

Version 1.1.0.0 (September 15, 2024)

Jinliang Wang

Institute of Zoology, Zoological Society of London, London NW1 4RY, UK

CONTENTS

1 Introduction	3
1.1 Overview	3
1.2 Features of the methods and software	3
1.3 Methods implemented in the software	3
1.4 Bug report	6
2 Installation	6
2.1 Windows version	6
2.2 Mac and Linux versions	7
2.3 Example dataset	7
3 Input Data Files: Empirical Analysis	7
3.1 No Windows GUI front end	7
3.2 Windows GUI front end	10
4 Running EMIBD9	12
4.1 No Windows GUI front end	12
4.2 Windows GUI front end	13
5 Output Files	13
6 Output in GUI front end	14
7 Simulations in GUI front end	17
8 Literature	19

1. Introduction

1.1 Overview

The computer program **EMIBD9** implements 2 likelihood methods to estimate the 9 condensed IBD coefficients, $\Delta = \{\Delta_1, \Delta_2, \dots, \Delta_9\}$, for a pair of individuals from their genotype data. Inbreeding coefficients of and relatedness (kinship coefficient) between individuals are then calculated from the estimated Δ . One method is designed to apply to a small sample or a sample containing a high proportion of close relatives where allele frequencies and their powers are poorly estimated by assuming a large unstructured sample (i.e. a sample of non-inbred and unrelated individuals). It adopts an expectation maximization (EM) algorithm to estimate both Δ and allele frequencies jointly. The other method is designed to apply to a sample of individuals containing a low proportion of close relatives. It is fast because it estimates Δ only and does not update allele frequencies by the EM algorithm.

The software package **EMIBD9** runs on Windows, Mac and linux computers. It includes the source code, executable, user's guide and example datasets. The computational part of **EMIBD9** program was written in Fortran 2003, and the source code can be compiled and run in different platforms (Windows, Linux and Mac). The Windows version of **EMIBD9** also includes a front end written in Visual Basic to help preparing input data and parameters and viewing analysis results in tables and graphs. The front end can also be used to simulate genotype data for individuals with given relationships. The simulated data can then be analysed by **EMIBD9** to check the estimation accuracy and data sufficiency.

1.2 Features of the method and software

The current version (V1.1.0.0) of **EMIBD9** has the following features:

- Simulating genotype data for individuals with user defined relationships, using user defined marker number and allele frequencies;
- Estimating 9 IBD coefficients of each pair of individuals, calculating relatedness and inbreeding coefficients from the estimated IBD coefficients;
- Allowing parallel runs using multiple cpus/cores by MPI and openMP to speed up the analysis of large datasets;
- Allowing many loci (~1 million SNPs) and many individuals (~30k) for the likelihood method without updating allele frequencies;
- Windows GUI.

The methods implemented by the software are introduced in the following paper:

Wang, J. 2022. A joint likelihood estimator of relatedness and allele frequencies from a small sample of individuals. *Methods in Ecology and Evolution* 13 (11), 2443-2462.

1.3 Methods implemented in the software

There are 9 possible IBD configurations between the two alleles of a diploid individual X and the two alleles of a diploid individual Y at a locus, as depicted in Figure 1 (Jacquard 1972). Configurations D_i ($i=1, 2, \dots, 9$) are random variables, taking values 1 or 0 with probabilities Δ_i and $1 - \Delta_i$, respectively. Because any two genotypes must fall into one of these 9 exclusive configurations, we have $\sum_{i=1}^9 \Delta_i = 1$. Given $\Delta = \{\Delta_1, \Delta_2, \dots, \Delta_9\}$, the inbreeding coefficient of X is $F_X = \Delta_1 + \Delta_2 + \Delta_3 + \Delta_4$, and that of Y is $F_Y = \Delta_1 + \Delta_2 + \Delta_5 + \Delta_6$. The coancestry coefficient or coefficient of kinship between X and Y is $\theta_{XY} = \theta_{YX} = \Delta_1 + \frac{1}{2}(\Delta_3 + \Delta_5 + \Delta_7) + \frac{1}{4}\Delta_8$. Wright's (1922) coefficient of relationship, R_{XY} , defined by the proportion of genes that X and Y have in common as a result of their genetic relationship, is $R_{XY} =$

$\frac{2\Delta_1 + \Delta_3 + \Delta_5 + \Delta_7 + \frac{1}{2}\Delta_8}{2\sqrt{(1+F_X)(1+F_Y)}} = \frac{2\theta_{XY}}{2\sqrt{(1+F_X)(1+F_Y)}}$. For a population without inbreeding, we have $\Delta_7 + \Delta_8 + \Delta_9 \equiv 1$ and $\Delta_1 = \Delta_2 = \Delta_3 = \Delta_4 = \Delta_5 = \Delta_6 = 0$.

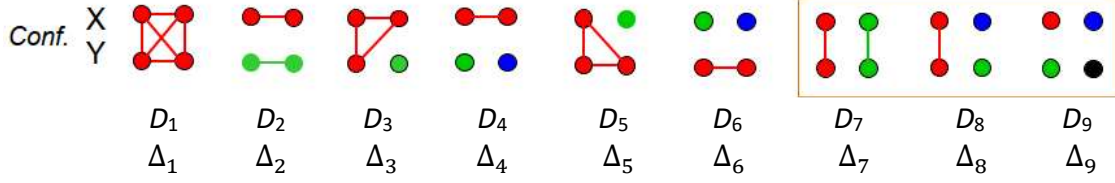


Figure 1. The 9 condensed IBD configurations (denoted by $\mathbf{D} = \{D_1, D_2, \dots, D_9\}$) and their corresponding probabilities (denoted by $\Delta = \{\Delta_1, \Delta_2, \dots, \Delta_9\}$) for a pair of individuals X and Y. In each configuration, each of the two diploid individuals X and Y has 2 genes shown as coloured disks horizontally. Within or between individuals, genes (disks) that are IBD are in the same colour and linked by lines, while genes that are not IBD are in different colours and unlinked by lines. Without inbreeding, only $\{D_7, D_8, D_9\}$ in the orange box are possible.

The IBD coefficients and coancestry (θ) of some common relationships are listed in Table 1.

Table 1 IBD coefficients and coancestry of some common relationships

Kinship	Δ_1	Δ_2	Δ_3	Δ_4	Δ_5	Δ_6	Δ_7	Δ_8	Δ_9	θ
Fullsibs whose parents are fullsibs	1/16	1/32	1/8	1/32	1/8	1/32	7/32	5/16	1/16	3/8
Full sibs	0	0	0	0	0	0	1/4	1/2	1/4	1/4
Parent-offspring	0	0	0	0	0	0	0	1	0	1/4
Half sibs	0	0	0	0	0	0	0	1/2	1/2	1/8
First cousin	0	0	0	0	0	0	0	1/4	3/4	1/16
Double first cousin	0	0	0	0	0	0	1/16	6/16	9/16	1/8
Second cousin	0	0	0	0	0	0	0	1/16	5/16	1/64
Avuncular	0	0	0	0	0	0	0	1/2	1/2	1/8

$\Delta = \{\Delta_1, \Delta_2, \Delta_3, \Delta_4, \Delta_5, \Delta_6, \Delta_7, \Delta_8, \Delta_9\}$ describes the genetic structure of individuals X and Y fully. It can be used to calculate inbreeding coefficients of and relatedness between individuals. Unfortunately, however, IBD modes, $\mathbf{D} = \{D_1, D_2, \dots, D_9\}$, are unobservable and their probabilities Δ are unknown. What we can do is to infer Δ from the marker genotypes of X and Y, or more specifically from the allele sharing or identity in state (IIS) patterns observed between genotypes of X and Y. For a locus with 4 or more alleles, there are 9 exclusive IIS modes (Table 2), $\mathbf{S} = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9\}$. However, for a locus with 2 alleles such as SNPs, modes S_4, S_6, S_8 , and S_9 are impossible. For a locus with 3 alleles, mode S_9 is impossible.

The probability of a given observed IIS mode, S_i ($i=1, 2, \dots, 9$), is a function of Δ and the allele frequencies, $\mathbf{p} = \{p_1, p_2, \dots, p_k\}$, where k is the number of alleles at the locus (Jacquard 1972). These probabilities are listed in Table 2. Suppose X and Y are genotyped at L loci, so that we have L observed IIS configurations (modes). Given the allele frequencies of the L loci and assuming linkage equilibrium, the probability of observing the L observed IIS configurations of X and Y is simply the product of the probability of observing each IIS mode as listed in Table 2. Maximizing this probability with respect to Δ leads to maximum likelihood estimates of Δ .

TABLE 2 Probability of genotype identity-in-state modes S_i given identity-by-descent modes D_i

IIS mode	Allelic state	IBD modes								
		D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9
S_1	$A_i A_i, A_i A_i$	p_i	p_i^2	p_i^2	p_i^3	p_i^2	p_i^3	p_i^2	p_i^3	p_i^4
S_2	$A_i A_i, A_j A_j$	0	$p_i p_j$	0	$p_i p_j^2$	0	$p_i^2 p_j$	0	0	$p_i^2 p_j^2$
S_3	$A_i A_i, A_i A_j$	0	0	$p_i p_j$	$2p_i^2 p_j$	0	0	0	$p_i^2 p_j$	$2p_i^3 p_j$
S_4	$A_i A_i, A_j A_k$	0	0	0	$2p_i p_j p_k$	0	0	0	0	$2p_i^2 p_j p_k$
S_5	$A_i A_j, A_i A_i$	0	0	0	0	$p_i p_j$	$2p_i^2 p_j$	0	$p_i^2 p_j$	$2p_i^3 p_j$
S_6	$A_j A_k, A_i A_i$	0	0	0	0	0	$2p_i p_j p_k$	0	0	$2p_i^2 p_j p_k$
S_7	$A_i A_j, A_i A_j$	0	0	0	0	0	0	$2p_i p_j$	$p_i p_j (p_i + p_j)$	$4p_i^2 p_j^2$
S_8	$A_i A_j, A_i A_k$	0	0	0	0	0	0	0	$p_i p_j p_k$	$4p_i^2 p_j p_k$
S_9	$A_i A_j, A_k A_l$	0	0	0	0	0	0	0	0	$4p_i p_j p_k p_l$

Alleles with different subscripts are distinct.

The 2 likelihood estimators implemented in the software are briefly described below.

(1) \hat{r}_{EM1}

Allele frequencies are commonly assumed known in deriving moment or likelihood estimators of IBD coefficients or relatedness. In reality, however, allele frequencies are seldom known but have to be estimated from the same sample of individuals whose relatedness is being estimated. Invariably it is assumed a sample is taken from a large random mating population such that all sampled individuals are noninbred and unrelated. Under this assumption, allele frequencies can be estimated simply by allele counting from the genotype data. Unfortunately, the above assumption is probably never true and that is exactly why we are estimating the inbreeding of and relatedness between sampled individuals.

Even when the assumption is true such that allele frequencies are unbiasedly estimated, the powers and products of allele frequencies are still poorly estimated except when sample size is large. It is however these powers and products of allele frequencies that are used in any moment or likelihood estimators, which become biased inevitably if these powers and products are biased. Suppose c_1 copies of allele A_1 are observed in the N sampled genotypes at a locus. Assuming the absence of genetic structure, $\hat{p}_1 = c_1/(2N)$ provides the best unbiased (i.e. $E(\hat{p}_1) = p_1$) estimate of p_1 . However, $\hat{p}_1^m = (c_1/(2N))^m$ is a poor estimate of p_1^m when $m > 1$. It overestimates p_1^m in expectation, with the overestimation increasing with a decreasing sample size N . An unbiased estimator p_1^m is (Wang 2021)

$$\hat{p}_i^m = \prod_{n=0}^{m-1} \frac{c_i - n}{2N - n}, \quad (m = 1 \sim 4)$$

The likelihood estimator, \hat{r}_{EM1} , was developed to estimate allele frequencies \mathbf{p} and IBD coefficients $\mathbf{\Delta}$ (and thus relatedness and inbreeding coefficients) jointly from the genotype data (Wang 2022). Iteratively, allele frequencies, their powers and products are estimated from the genotype data and the estimated IBD coefficients among all individuals in the sample, accounting for the small sample size. These estimates are then used in updating IBD coefficients ($\mathbf{\Delta}$) for all pairs of individuals in the sample. Iterating between the two alternative types of updates to convergence in the expectation-maximization algorithm leads to maximum likelihood estimates of both $\mathbf{\Delta}$ and \mathbf{p} .

(2) \hat{r}_{EM2}

\hat{r}_{EM1} is expensive to calculate, because the number of variables to be estimated jointly (allele frequencies at all loci and $9N^2$ IBD coefficients) when the number of individuals N or the number of loci L is large. For a sample containing a small proportion of close relatives, updating allele frequencies has little benefit but costs a lot of computation. In such a case, I use \hat{r}_{EM2} to estimate Δ of each pair of individuals separately, accounting for small N but not updating allele frequencies. Therefore \hat{r}_{EM2} is fast to compute, and as a result can be applied to a large sample of individuals (N) and many loci (L).

Details of the two likelihood methodologies and their performance in comparisons with other methods can be found in Wang (2022), which should be cited when using the **EMIBD9** software. In this document, I will focus on how to use **EMIBD9** to analyse an empirical dataset, how to conduct a simulation, and how to understand and interpret the analysis results.

1.4 Bug report

EMIBD9 (Copyright 2021 by Jinliang Wang) is available, free of charge, for academic use only. It is downloadable from the website <http://www.zsl.org/science/research/software/>. Any updates of the program will also be put in the same website. Every effort has been made to implement the methods correctly and efficiently, but there is no guarantee that the program is free of bugs. Reports of bugs are welcome, and should be sent to: <mailto:jinliang.wang@ioz.ac.uk?subject=EMIBD9>.

2. Installation

EMIBD9 is written in Fortran 2003, and is compiled for Windows 10, Mac and Linux 64bit operating systems. For Windows, it also has a graphical user interface (GUI) written in Vb.net, which can be used to help inputting data and analysis parameters, and viewing analysis results in graphs and tables. Mac users can also install a Windows simulator to run the Windows version, as the Mac version is x-terminal based and has no GUI. For large genomic data, it is better to use the linux version for MPI and openMP parallel computation using many cores of a linux cluster.

2.1 Windows version

For Windows users, please download and unzip the zipped file for Windows version of **EMIBD9** to obtain an installation file called “EMIBD9Setup.msi”. Double click this file to start the installation. By default, it will be installed in “C:\ZSL\EMIBD9”. However, you can change the directory where **EMIBD9** will be installed during the installation process. It is suggested that it NOT be installed in the “Windows” directory or the “Program Files” directory. Otherwise, due to windows security issues, subsequent input and output files of **EMIBD9** might be automatically moved to a folder in VirtualStore and, the simulation program may not run properly.

Upon installation, you will find **EMIBD9** executables, user’s manual in PDF, libraries, and an example dataset in your **EMIBD9** program folder. A shortcut to **EMIBD9** program is also placed in your Windows’ desktop. Double clicking the shortcut will start the program.

EMIBD9 should run on a PC or server with Microsoft Windows operating system version 10, 64bit. It requires the .Net Framework 4.5.2 (or higher), which is probably already installed on your computer. You can check by clicking Start on your Windows desktop, selecting Control Panel, and then double-clicking the Add or Remove Programs icon. When that window appears, scroll through the list of applications to check whether Microsoft .Net Framework 4.5.2 (or higher) is listed. If it is not installed

(which is unlikely), you need first download it from the Microsoft website and install it before installing **EMIBD9**. Occasionally, .Net might be installed but not enabled on your computer. In such a case, you need to enable it for **EMIBD9**'s GUI front end to work.

This PDF file of the user's manual is included in the package. I suggest opening/printing and reading this document before running **EMIBD9**. The analysis on one's own dataset using the program is described below.

2.2 Mac and Linux versions

For Mac and Linux, just download the corresponding package, unzip it, and then copy everything (including the folders, subfolders and files) to your desired location (more details in file, readme.pdf, in the downloaded package). Everything described above, except for the GUI and the simulation program, should be included in the package.

It is suggested to include the **EMIBD9** program path permanently in the automatic search paths of your shell so that the program can be launched conveniently without specifying the path of the program. If you use the Bash shell of linux, for example, you can add the line "export PATH=\$PATH:prmpath" (where prmpath is the path for **EMIBD9** program folder where the binaries are found) to file .bashrc that will be read when your shell launches. To do so, simply type "nano ~/. bashrc", append the line, and save the file. The next time you launch your shell, the shell knows where to find **EMIBD9** program (e.g. EM_IBD_P) to run an analysis, no matter where (in which folder) the command line invoking EM_IBD_P is issued. If you use zsh in Mac, append "export PATH=\$PATH:prmpath" to file .zshenv. Similarly, if you run **EMIBD9** in DOS, you need to add **EMIBD9** program folder to the environment variable by using "Control Panel>System>Advanced System Settings>Environment Variables".

2.3 Example dataset

An example dataset is also installed in **EMIBD9** program folder. The example "ant" subfolder has 2 files used for setting up an empirical data analysis. The "readme.txt" file briefly describes the parameters to be used in setting up the project, and the other file is the genotype data file.

3 Input Data Files: Empirical Analysis

For a **EMIBD9** analysis of an empirical dataset, two input files in specific formats need to be prepared. One file, called parameter file hereafter, contains analysis parameters. The other one, called data file hereafter, contains genotypes of sampled individuals. I describe how to prepare the two files for running **EMIBD9** with Windows GUI front end (on a PC running Windows 10) and without (on Windows DOS or on the x-terminal of Linux, Mac).

3.1 No Windows GUI front end

To keep input and output files well organized, it is advised to create a suitably named (a string with no space and no other illegal characters for file/folder names) new folder as your project folder, and save the data file and parameter file in this folder. The path of the project folder, called *project path* hereafter, will be frequently used in working with the project.

3.1.1 Parameter file

A parameter file with a suitable name (say, MyData.par) needs to be prepared and saved in the project folder. The extension name ".par" is not compulsory, but is recommended as in Windows GUI it has

this fixed extension “.par”. The file should have exactly 10 text lines in the right order (below), with each line listing a single parameter value. In the example file shown below, each line is started with a parameter value, followed by the comment/note (the part following exclamation mark !). In the comment, the 1st, 2nd and 3rd parts (separated by commas) are the parameter type, name, and meaning (where “#” means “number of”), respectively. Note, a Boolean type variable takes an integer value 1 or 0 for TRUE or FALSE, respectively. Also note that for each line, only the parameter value is necessary, and the comment is just for clarity. More details about each parameter in each line are as follows.

```

377          !Integer, NumIndiv,      #Individuals
6           !Integer, NumLoci,       #Loci
0           !Integer, DataForm,      0/1/2
1           !Boolean, Inbreed,       Inbreeding: 1/0=T/F
D:\EMIBD9\Ant\ant377.dat !String, GtypeFile,      Gtype file path & name
D:\EMIBD9\Ant\ant377.ibd9 !String, OutFileName,     output file path & name
111         !Integer, ISeed,         Random number seed
0           !Boolean, RndDelta0,     1/0=T/F
1           !Boolean, EM_Method,     0/1=Update ( $\Delta$  only)/Update ( $\Delta$  &  $p$ ) jointly
0           !Boolean, OutAlleleFre, 1/0=T/F

```

1. NumIndiv: An integer, the number of sampled and genotyped individuals to be analysed for IBD. An individual with no or missing genotype data at all loci has no information and should be excluded from an analysis. Individuals are assumed diploid at all marker loci.

2. NumLoci: An integer, the number of loci genotyped for each of **NumIndiv** individuals. All loci are assumed codominant. Monomorphic loci are allowed but will be identified and excluded by the program in the analysis. A locus can have a maximum of 127 alleles.

3. DataForm: An integer, taking possible values 0, 1, and 2 to indicate genotype data format. When **DataForm**=0, the genotypes of an individual are listed in a single row, with columns $2l-1$ and $2l$ listing the 1st and 2nd alleles at locus l ($=1, 2, \dots, L$). The individual identifier (a string of a maximal length 20 characters) is listed on column 0. Therefore, there are $2L+1$ columns on a single row for each individual. Separators between columns are blank spaces. When **DataForm**=1, the identifier of individual i ($=1, 2, \dots, N$) is listed in the i th column of row 0. For locus l ($=1, 2, \dots, L$), the genotype of individual i is listed in column i of row l , taking a value of 0, 1 or 2 denoting the number of reference alleles in the genotype. A missing genotype is denoted by 3. There should be no separators between genotypes. When **DataForm**=2, everything is the same as **DataForm**=1, except the genotype matrix $L \times N$ is transposed. Note while **DataForm**=0 can be used for markers with any number of alleles at a locus, **DataForm**=1 and **DataForm**=2 only apply to markers with 2 or fewer alleles at each locus. Because only 4 values, 0, 1, 2 and 3, are possible, 2 bits can be used to store the genotype data at a diallelic locus of an individual. A 4-byte integer can be used to store genotype data at 16 loci. Therefore, data in **DataForm**=1 and **DataForm**=2 take much less space in data file, can be read faster, and are stored with much less RAM.

4. Inbreed: A Boolean, taking values 0 or 1 to indicate inbreeding is not and is allowed in estimating IBD coefficients. With **Inbreed**=0, $\Delta_1 = \Delta_2 = \Delta_3 = \Delta_4 = \Delta_5 = \Delta_6 = 0$ is assumed for each dyad, and thus $F_X = 0$ for any individual X . With **Inbreed**=1, all 9 condensed IBD coefficients are estimated for each dyad.

5. **GtypeFile**: A string, giving the path and name of the genotype file. The string has a maximal number of 499 characters, and should not contain illegal characters for file/folder names (such as “/” or “\”).
6. **OutFileName**: A string, giving the path and name of the output file to be produced by **EMIBD9**. The same restriction to string **GtypeFile** applies to **OutFileName**.
7. **ISeed**: An integer used to seed the random number generator.
8. **RndDelta0**: A Boolean, taking values 0 or 1. **RndDelta0** takes effect only when **EM_Method** =1 (below) such that both **p** and **Δ** are jointly estimated. In such a case, initial values for **Δ** can be set in two possible ways. One (**RndDelta0**=0) is to give values taken at random from a uniform distribution in the range [0,1] to Δ_j ($j=1, 2, \dots, 9$) for each dyad. These initial values are normalized by $\Delta_j = \Delta_j / \sum_{i=1}^9 \Delta_i$. Alternatively (**RndDelta0**=1), random initial values are used for **Δ** to obtain the maximum likelihood estimates (MLE) of **Δ** by estimator \hat{r}_{EM2} (assuming **EM_Method** =0). Then these MLE of **Δ** are used as initial values by estimator \hat{r}_{EM1} in the EM algorithm for updating both **p** and **Δ**.
9. **EM_Method**: A Boolean, taking values 0 or 1 to indicate whether to estimate **Δ** only (**EM_Method**=0) or to estimate **Δ** and **p** jointly (**EM_Method**=1).
10. **OutAlleleFre**: A Boolean, taking values 0 or 1 to indicate whether to output estimated allele frequencies (=1) or not (=0).

3.1.2 Data file

A data file named in the parameter file above should be prepared and saved in the project folder. It contains the ID and genotypes at each locus for each sampled individual. When markers are all diallelic, any genotype data format is accepted. Otherwise, only the first data format (individual data in 1 row) is accepted.

1. Individual data in 1 row

This data format is used for both multiallelic and diallelic marker data. In the parameter file, parameter **DataForm** must take value 0.

All data for an individual are placed in a single row, with columns separated by a single blank space. The possible columns are as follows.

IndivID: The 1st column is individual identifier, **IndivID**, which is compulsory. It is a string of a maximal length of 20 characters. The string must NOT contain blank space and other illegal characters (such as /), and must be unique among all sampled individuals (i.e. NO duplications). Any string longer than 20 characters for individual ID will be truncated to have 20 characters.

Genotype: From the 2nd column on, two consecutive columns list the two observed alleles (represented by two integers, each of the range [0, 32767]) at a locus, separated by a single blank space. A missing allele at any locus is denoted by 0, and a missing diploid genotype is denoted by 0 0. Data for two example individuals at 3 loci are as follows.

```
Bob 110 116 142 148 120 126
Peter 110 120 120 124 150 154
```

2. Individual genotypes in columns

This data format is for diallelic markers only, with **DataForm=1** in the parameter file.

The zeroth row lists individual identifiers, **IndivID**, for individual 1, 2, ..., N . Identifiers are separated by a blank space.

The l th row lists genotypes for marker locus l ($=1, 2, \dots, L$). In the row, the i th column lists the genotype (0,1,2,3) for individual i ($=1, 2, \dots, N$). A genotype is encoded by a number 0, 1 or 2, the number of copies of the (arbitrary) reference allele in the genotype. A missing genotype is encoded by 3. There is NO separator between genotypes.

Data for two example individuals (5 SNPs) are as follows.

```
Bob Peter
12
10
01
10
31
```

3. Individual genotypes in rows

This data format is for diallelic markers only, with **DataForm=2** in the parameter file.

The zeroth row lists individual identifiers, **IndivID**, for individual 1, 2, ..., N . Identifiers are separated by a blank space.

The i th row lists genotypes for individual i ($=1, 2, \dots, N$). In the row, the l th column lists the genotype (0,1,2,3) for locus l ($=1, 2, \dots, L$). A genotype is encoded by a number 0, 1 or 2, the number of copies of the (arbitrary) reference allele in the genotype. A missing genotype is encoded by 3. There is NO separator between genotypes.

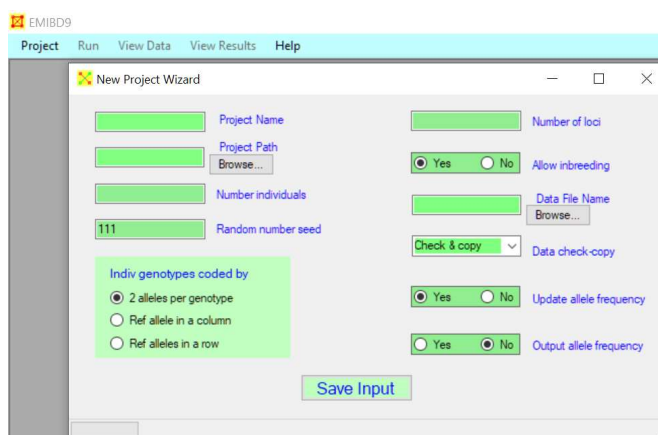
Data for two example individuals (5 SNPs) are as follows.

```
Bob Peter
11013
20101
```

3.2 Windows GUI front end

3.2.1 Parameter file

The same parameter file as described in “**3.1.1 Parameter file**” is prepared more conveniently by using **EMIBD9**’s GUI front end (Figure 2). The new project wizard is started by clicking “Project” and then “New Project”.



1. *Project Name*: The text box accepts a string that will be used as output file names and the project folder name as well. On completing parameter input using the wizard, a parameter file *.par and a data file *.dat will be written to the project folder, where * is Project Name provided in this text box. In the parameter file, the output file name is *.ibd9.
2. *Project Path*: The text box accepts a string specifying the path of the project. A project folder with the above specified project name will be set up in the project path specified. All output files, on completing an **EMIBD9** analysis, will be written into this project folder. You can also use the Browse... button to locate the path where to make the project folder (path).
3. *# individuals*: An integer giving the value of **NumIndiv**, number of sampled individuals.
4. *# loci*: An integer giving the value of **NumLoci**, number of sampled loci.
5. *Allow Inbreeding*: Radio buttons to choose values for **Inbreed**, whether to allow inbreeding in IBD estimation or not.
6. *Random Number Seed*: An integer giving the value of **ISeed**, seed for random number generator.
7. *Data File Name*: A string specifying the path and name of the data file. You can also use the Browse... button to locate the file.
8. *Data Check-Copy*: There are 3 options for dealing with genotype data. The 1st option is “No Check & Copy”, which means genotype data will not be checked (for format and validity) and will not be copied to the project folder. This option is useful when the data file is large and you do not want to duplicate it in your computer. The file path and name will be provided in the parameter file and **EMIBD9** will read the file from its original location. The 2nd option is “Copy”, which means genotype data will be copied to the project folder but not checked. The 3rd option is “Check & Copy”, which means genotype data will be checked and copied to the project.
9. *Genotype data coded*: Three radio buttons are used for the 3 options for data formats. “2 alleles per genotype” means **DataForm**=0. “Ref allele in a column” and “Ref allele in a row” mean **DataForm**=1 and **DataForm**=2, respectively.
10. *Update allele frequency*: Checking “Yes” radio button means **EM_Method**=1, and checking “No” radio button means **EM_Method**=0.

11. *Output allele frequency*: Checking “Yes” radio button means **OutAlleleFre**=1, and checking “No” radio button means **OutAlleleFre**=0.

3.2.2 Data file

A data file with format specified by **DataForm** should be prepared with name specified in “7. *Data File Name*”. **EMIBD9** will deal with this file, with rather limited check of the format and validity of the data if the option “Check & Copy” is chosen. There is no guarantee that runtime errors will not occur due to data errors. The data may or may not be copied to the project folder, depending on the options chosen in “8. *Data Check-Copy*”.

4 Running EMIBD9

Once valid parameter file and data file have been prepared, you can call **EMIBD9** to estimate IBD coefficients, inbreeding and kinship coefficients (relatedness).

4.1 No Windows GUI front end

1. Open a DOS window (for PC, using Windows command cmd.exe), or a Linux’ or Mac’s x-terminal. For the latter 2 cases, it is better to get the administrator’s privileges (e.g. using “sudo -s” on Mac) to run the commands smoothly without permission problems.
2. Navigate to your project folder (where the parameter file *.par is found) by using command “cd”.
3. Start a serial run by the following command line

```
mypath\EM_IBD_P INP:MyData.par (for DOS)
mypath/EM_IBD_P INP:MyData.par (for Mac)
mypath/EM_IBD_P INP:MyData.par (for Linux)
```

where “mypath” is the path of the binary EM_IBD_P and “MyData.par” is the name of the parameter file (which should be in the project folder), “INP:” is the command line flag for input (INP) file.

If you have already set the **EMIBD9** program folder in the environment variable of your computer after installation of the program, you can navigate to (using command cd) your **EMIBD9** project folder and start a serial run by command line

```
EM_IBD_P INP:MyData.par (for DOS)
EM_IBD_P INP:MyData.par (for Mac)
EM_IBD_P INP:MyData.par (for Linux)
```

4. Alternatively start an MPI run with M parallel threads by the following command line (started in your project folder)

```
mpiexec -n M mypath\EM_IBD_P INP:MyData.par (for DOS)
mpirun -n M mypath/EM_IBD_P_mpi INP:MyData.par (for Mac)
mpirun -n M mypath/EM_IBD_P_mpi INP:MyData.par (for Linux)
```

If the **EMIBD9** program folder is in your computer’s the environment variable, you can navigate to your project folder and launch MPI run with command line

```
mpiexec -n M EM_IBD_P INP:MyData.par (for DOS)
mpirun -n M EM_IBD_P_mpi INP:MyData.par (for Mac)
mpirun -n M EM_IBD_P_mpi INP:MyData.par (for Linux)
```

Here are the command line flags which can be used in running **EMIBD9** to override the corresponding parameter values set in a parameter file.

- (1) “IND:[Number of individuals]”: The number of individuals, **NumIndiv**. It must be equal to or smaller than the number of individuals with genotypes in data file. In the latter case, only data for the first **NumIndiv** individuals in the data file will be analysed.
- (2) “LOC:[Number of loci]”: The number of loci, **NumLoci**. It must be equal to or smaller than the number of markers genotyped for an individual in data file. In the latter case, only genotype data for the first **NumLoci** loci in the data file will be analysed.
- (3) “RAN:[Random number seed]”: The seed for random number generator, **ISeed**.
- (4) “INP:[Input parameter file name]”: The path and name for the parameter file. If starting the run from your project folder, it is unnecessary to give the path for parameter file. Just the name is fine.
- (5) “OUT:[Output file name]”: The path and name of output files, the value of **OutFile**. If starting the run from your project folder, it is unnecessary to give the path of the output file.
- (6) “GEN:[Genotype file name]”: The path and name of genotype file, the value of **DataName**. If starting the run from your project folder, it is unnecessary to give the path for genotype file.
- (7) “UPP:[Update allele frequency (0/1)]”: Option “UPP:1” means updating allele frequencies, i.e. **EM_Method**=1. Option “UPP:0” means NOT updating allele frequencies, i.e. **EM_Method**=0.
- (8) “OMP:[Number of openMP threads per MPI process]”: It is the number of openMP threads to be used per MPI process. You can also use multiple openMP threads without using MPI (i.e. a single process).

An example command line in DOS (started in project folder) is

```
mpiexec -n 4 mypath\EM_IBD_P INP:MyData.par IND:99 LOC:9 RAN:222 OUT:mytest1  
GEN:myGtype.dat UPP:1
```

Note these flags can appear in any order AFTER the **EMIBD9** program name, and can be used in any combinations. Also note these flags have exactly 3 capital letters followed by colon “:” (there is no blank space before or after :).

To show these flags, you can type “EM_IBD_P help” in DOS, “./EM_IBD_P help” in Mac’s X terminal, and “./EM_IBD_P help” in linux’s X-terminal.

4.2 Windows GUI front end

The GUI asks you to provide a number of MPI parallel processes to be used in analysing the data. The minimum value is 1, which specifies actually a serial run. The maximum is suggested by **EMIBD9**, which is the number of hyperthreads of your computer. Selecting a value less than 1 is illegal, and a value larger than the suggested maximum value has no gains in efficiency. Runtime parameters that could be set with no Windows GUI front end as described above cannot be changed at runtime; they are specified in the parameter file and will be used in analysis.

5 Output Files

A single output file, as named in the parameter file, will be generated in the project folder. It has the following information.

5.1 Run parameters and basic data statistics

This section lists the main parameter values and running options used in conducting the analysis.

5.2 Running time, tolerance and iterates

This section lists the starting and finishing date and time of the analysis, the number of iterations and the final tolerance. It also lists the tolerance at each iteration.

5.3 IBD coefficient estimates

For a sample of N individuals, there are N^2 rows in this section, with each row lists the analysis results for each dyad. The 1st column gives the order of the dyad, starting from 1 which is the dyad of individual 1 with itself. The 2nd and 3rd columns list the two individuals of the dyad. The 4th to 12th columns list the number of loci at which the pair of genotypes are observed to have IIS mode S_1 to S_9 . The 13th to 21st columns list the estimated values of Δ_1 to Δ_9 . The last (22^{ed}) column lists the kinship (coancestry) coefficient of the dyad.

5.4 Individual inbreeding coefficients

This section has N rows, each individual taking one row. The 1st column gives the individual identifier, the 2nd and 3rd columns list the number of loci at which the individual has missing data and has complete genotype data. The 4th and the 5th columns list the number of loci at which the individual has homozygous and heterozygous genotypes, respectively. The 6th and the 7th columns give the average and the SD of the inbreeding coefficient estimates for the individual.

6 Output in GUI front end

The output can be viewed in tables and graphs as well as texts in **EMIBD9**'s GUI front end. The graphs can be sized, and saved to files or copied to clipboard for pasting (Ctrl V) to a document. For the interpretations of the outputs, please see section 5 above.

6.1 IIS

The numbers of loci at which genotypes of a dyad are in IIS mode i , S_i (or IIS_i) for ($i=1, 2, \dots, 9$), can be viewed in a table by clicking "View Results", "#IIS". An example is shown below.

EMIBD9 (Project=HGDP) - [Observed #IIS across loci per dyad]

Dyad#	Indiv1	Indiv2	IIS1	IIS2	IIS3	IIS4	IIS5	IIS6	IIS7	IIS8	IIS9	Missing
1	HGDP01405	HGDP01405	450979	0	0	0	0	0	192326	0	0	953
2	HGDP01405	HGDP01406	293246	37346	120181	0	118624	0	73584	0	0	1277
3	HGDP01405	HGDP01408	293222	37641	119455	0	117913	0	74035	0	0	1992
4	HGDP01405	HGDP01411	290223	38230	121847	0	117329	0	74685	0	0	1944
5	HGDP01405	HGDP01412	294467	37924	118230	0	118995	0	73167	0	0	1475
6	HGDP01405	HGDP01413	290260	38273	121716	0	118042	0	73951	0	0	2016
7	HGDP01405	HGDP01414	293994	37341	119169	0	119184	0	72934	0	0	1636
8	HGDP01405	HGDP01415	288612	39537	122541	0	118482	0	73710	0	0	1376

6.2 IBD coefficients in a table

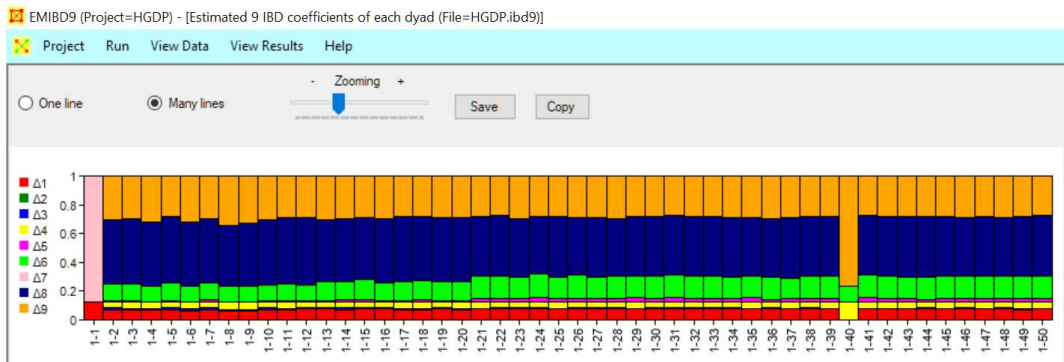
The estimated values of $\Delta=\{\Delta_1, \Delta_2, \Delta_3, \Delta_4, \Delta_5, \Delta_6, \Delta_7, \Delta_8, \Delta_9\}$ and the value of θ (denoted by $r(1,2)$) calculated from Δ can be viewed in a table by clicking "View Results", "IBD9 and r ", "Table". An example is shown below.

EMIBD9 (Project=HGDP) - [Estimated IBD and r]

Dyad#	Indiv1	Indiv2	Δ_1	Δ_2	Δ_3	Δ_4	Δ_5	Δ_6	Δ_7	Δ_8	Δ_9	r(1,2)
1	HGDP01405	HGDP01405	0.1230	0.0000	0.0000	0.0000	0.0000	0.0000	0.8770	0.0000	0.0000	0.5615
2	HGDP01405	HGDP01406	0.0684	0.0000	0.0144	0.0407	0.0124	0.1122	0.0000	0.4482	0.3037	0.1939
3	HGDP01405	HGDP01408	0.0704	0.0000	0.0084	0.0445	0.0124	0.1114	0.0000	0.4536	0.2994	0.1942
4	HGDP01405	HGDP01411	0.0671	0.0000	0.0090	0.0470	0.0023	0.1069	0.0000	0.4446	0.3230	0.1839
5	HGDP01405	HGDP01412	0.0695	0.0000	0.0128	0.0409	0.0085	0.1251	0.0000	0.4611	0.2821	0.1954
6	HGDP01405	HGDP01413	0.0643	0.0000	0.0108	0.0476	0.0051	0.1098	0.0000	0.4444	0.3179	0.1834
7	HGDP01405	HGDP01414	0.0670	0.0000	0.0173	0.0389	0.0163	0.1166	0.0000	0.4481	0.2958	0.1958
8	HGDP01405	HGDP01415	0.0646	0.0000	0.0085	0.0500	0.0013	0.1086	0.0000	0.4223	0.3448	0.1750

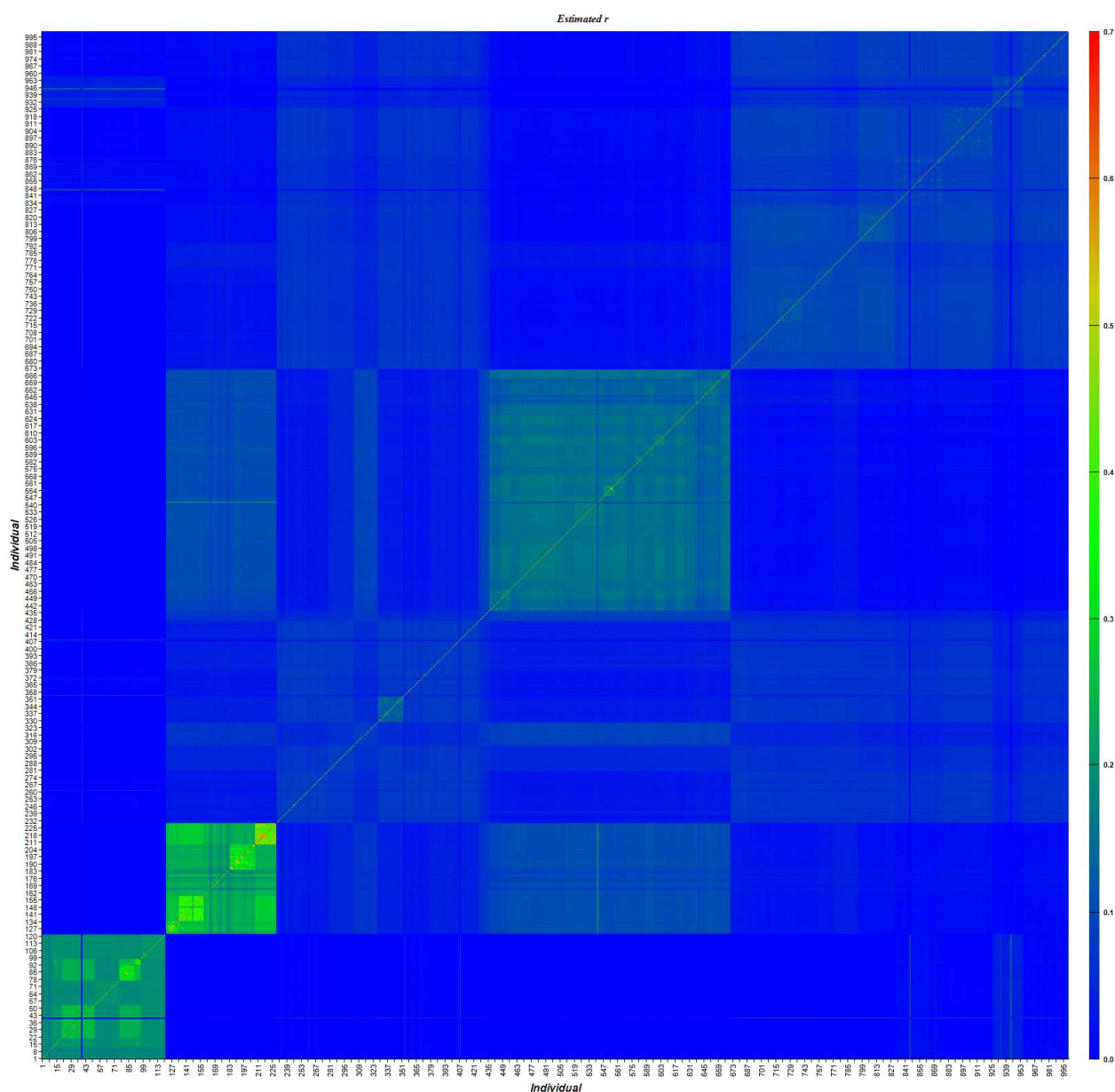
6.3 IBD coefficients in a bar chart

The estimated values of $\Delta = \{\Delta_1, \Delta_2, \Delta_3, \Delta_4, \Delta_5, \Delta_6, \Delta_7, \Delta_8, \Delta_9\}$ for each dyad can also be viewed in a bar chart by clicking “View Results”, “IBD9 and r ”, “ Δ_1 - Δ_9 bar chart”.



6.4 Δ_i heatmaps

The estimates of Δ_i ($i=1, 2, \dots, 9$) as well as r for each dyad can also be viewed in a heatmap. An example for r estimates is shown below.



6.5 Inbreeding coefficients bar chart

The estimated inbreeding coefficients for each individual can be viewed in a table or a bar chart, as shown below for an example.



7 Simulations in GUI front end

EMIBD9's GUI front end can also be used to simulate genotype data of individuals of a mixture of relationships. The simulated data can then be analysed by **EMIBD9** to see if the simulated genetic structure of a sample of individuals is reconstructed correctly or not. They can also be used in understanding the power and accuracy of **EMIBD9** and in evaluating data (sampling scheme) sufficiency, etc. With slight modification, the simulated data can also be analysed by other methods implemented in other software, such as COANCESTRY (Wang 2011). These methods can then be evaluated and compared using the same simulated data with known structures.

7.1 parameter input

By clicking “Project” and “New Simulation”, a new simulation project wizard shows up, as shown below. In the window, the wizard asks some parameters which are then written to a parameter file named “**SimuPara.txt**”. The file is saved in the project folder, and is used by the simulation program “SimIBD.exe” to generate simulated genotype data.

The information required by the new simulation project wizard is as follows.

(1) Family sizes

There are 8 types of families that can be simulated and included in a sample of individuals for IBD analysis. Family FSFS means full sibs whose parents are full sibs. Individuals of this familial

relationship have expected IBD coefficients as listed in Table 1. The textbox on the left side of “FSFS” accepts input of the number of individuals of this relationship to be simulated. The default value is 0 (i.e. no individuals of FSFS relationship are to be simulated), and you can change it to any natural numbers (1, 2, ...).

Similarly, family FS means full sibs, HS means half sibs who share the same parent of one sex but have different parents of the other sex. UR means unrelated individuals. PFS means parents with their children (full sibs). When the input for the number of PFS is less than 3, only that number of parents are generated. Otherwise, 2 parents with that number minus 2 full sibs are generated. PHS means one parent with its children (half sibs). When the input for the number of PHS is less than 2, only that number of parent is generated. Otherwise, 1 parent with that number minus 1 half sibs are also generated. FC means first cousins, and SC means second cousins.

(2) Number of loci

Numbers of loci in different allele frequency distribution types are accepted. “Uniform” means allele frequencies in a uniform distribution in the range [0, 1]. “Equal” means an equal frequency for each allele. For a locus with K alleles, the allele frequency would be $1/K$. “Triangular” means allele k ($=1, 2, \dots, K$) at a locus has a frequency proportional to $1/k$. “Triangular2” and “Triangular4” mean allele k ($=1, 2, \dots, K$) at a locus has a frequency proportional to $1/k^2$ and $1/k^4$, respectively. You can input a mixture of loci of different allele frequency distributions.

(3) Number of alleles

The number of alleles at each locus.

(4) Drop rate

Allelic dropout rate assumed in simulating genotype data. When a simulated genotype is a heterozygote, then a random number R is sampled from a uniform distribution in the range [0,1]. If R is smaller than drop rate, then an allelic dropout event will occur. In such a case, one of the two alleles in the genotype is selected at random and is regarded as invisible (drops out). The observed genotype would be the homozygote of the alternative allele.

(5) Other error rate

When other error rate is larger than 0, then each simulated allele in a genotype will be selected at random at that rate and will be changed to an allele chosen at random from all alleles present at the locus.

(6) Genotype format

Three options for genotype format are available, “2 alleles per genotype”, “Ref allele in a column” and “Ref allele in a row” which correspond to **DataForm=0**, **DataForm=1** and **DataForm=2** respectively.

(7) Data missing rate

The rate at which the genotype of any individual at any locus is missing. The simulated genotypes will be taken as missing at this rate.

(8) Random number seed

An integer used to seed the random number generator.

(9) Inbreeding

Whether to allow inbreeding in estimating IBD coefficients or not.

(10) Output allele frequency

Whether to output the estimated allele frequency to a file or not.

(11) Update allele frequency

Whether to update allele frequency (EM_Method=1, estimator \hat{r}_{EM1}) or not (EM_Method=0, estimator \hat{r}_{EM2}).

(12) Project path-name

The input for the path and name of the new simulation project.

7.2 Running simulation

On completing the input, you can click “Simulate” button to (1) check the inputs, (2) save the inputs to a file named “SimuPara.txt” in the project folder, and (3) simulate genotype data. If any errors were found in checking the input, a message will appear and you need to correct the inputs. If no errors were met, the simulated genotype data will be saved to a data file *.dat, the parameter values for running **EMIBD9** will be saved to a parameter file *.par, the simulated true IBD coefficients for each dyad will be saved to file *.TrueIBD, the simulated true inbreeding coefficients for each individual will be saved to file *.TrueInb, where “*” is the project name.

7.3 Simulation results

The simulated IBD coefficients for each dyad and the simulated inbreeding coefficients of each individual can be viewed in tables, in bar charts and in heatmaps.

8 Literature

Jacquard, A. 1972. Genetic information given by a relative. *Biometrics* 28: 1101–1114.

Wang, J. 2022. A joint likelihood estimator of relatedness and allele frequencies from a small sample of individuals. *Methods in Ecology and Evolution* 13 (11): 2443-2462.

Wang, J. 2011. COANCESTRY: a program for simulating, estimating and analysing relatedness and inbreeding coefficients. *Molecular Ecology Resources* 11: 141-5.

Wright, S. 1922. Coefficients of inbreeding and relationship. *Am Nat* 61: 330–338.