R_P_M: Reconstructing Pedigrees from Markers

Version 1.0

21st of April 2025

Jesús Fernández

GOBIERNO DE ESPAÑA

MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES

INIA
Instituto Nacional de Investigación y Tecnología Agraria y Alimentaria

INDEX

Introduction

In situations where pedigrees cannot be accurately recorded — whether in conservation or in commercial populations under breeding programs — genetic relationships can be directly calculated from molecular markers. However, when molecular information is scarce (as is often the case with wild populations, non-mainstream species or local breeds of livestock), these genetic relationships are not as accurate as those derived from pedigrees (Solberg et al., 2008; Gómez-Romano et al., 2013). In such cases, a better approach might involve reconstructing genealogies from molecular data, which requires less molecular information.

This new software tool for pedigree reconstruction (R_P_M) implements a method relying on the search for congruent genealogies where the pedigree coancestry matrix between genotyped individuals has the highest correlation with the molecular coancestry matrix (original approach developed in Fernández and Toro, 2006). The R_P_M software and its underlying algorithm offer high accuracy in pedigree reconstruction, even in situations with scarce molecular information (i.e., few markers with a low level of polymorphism). Additionally, it provides several advantages over currently implemented methods: it is robust against Hardy-Weinberg disequilibrium, independent of the knowledge of real allelic frequencies, capable of incorporating known restrictions (e.g., fixed or banned relationships), and, especially, able to explicitly include virtual individuals to complete the pedigree. Consequently, R_P_M serves as a valuable tool for studies on wild populations as well as for farm/domestic population managers.

Downloading and installation

Software can be downloaded from the GitHub repository jfmmvbb/R_P_M. Compiled executables are provided for Windows, Linux and Mac operating systems. In the folder, you will also find some example files (see details in the corresponding section) and this manual. Download all the files in your device and run it directly. Program will search for input files in the local folder, so they must be there if you want everything to run smoothly.

Algorithm

As said before, the methodology aims to identify a congruent genealogy that maximizes the correlation between the coancestry matrix derived from the reconstructed pedigree and the molecular coancestry obtained from genotypic data, as proposed by Fernández and Toro (2006). By default, the software calculates and uses as the reference for comparisons the similarity coancestry (i.e., Identity By State) described in Nejati-Javaremi et al. (1997). Nevertheless, a user-provided matrix can be used, recognising that many estimators of molecular coancestry have been proposed (see, for example, Morales-Gonzalez et al., 2020). Self-coancestries are not included in the calculations because preliminary results indicated higher accuracy when not accounting for the values in the diagonal. As both coancestry matrices (molecular and genealogical) are symmetric, only elements above the diagonal are used in the calculation, since correlation remained unchanged when reciprocal relationships are considered, thereby improving computational efficiency.

The optimisation (maximisation) process underlying the software's functionality is addressed using a *simulated annealing* algorithm (Kirkpatrick et al., 1983), which enables exploration across the space of feasible pedigree structures. The dynamics of the algorithm (and, thus, its efficiency) can be tuned using a list of parameters which includes the initial 'temperature' (related to the probability of acceptance of alternate solutions), the maximal number of steps (temperatures) to be simulated, the number of alternate solutions to generate in each step and the rate of 'cooling' (reduction in the temperature between steps). These parameters are provided to the programme using the file ***anneal_param.txt*** with this structure:

```
300  ! Maximum no. of steps ('temperatures')
10000  ! No. solutions proposed by step
3  ! Initial no. changes performed to create each alternate solution
.001  ! Initial 'temperature'
.9  ! Rate of 'cooling'
Y  ! Use a particular initial solution (Y) or not (N)
```

Hint: This file is required and its name is fixed. Some clues for modifying the values of the parameters in this file and, thus, improving the power of the algorithm can be found, for example, in Fernández and Toro (2006).

The last option of the parameters' file allows for defining an initial solution (i.e., pedigree) to start the search from (see section *Input files*/Initial solution for the particular file structure). This could be useful, for example, if the process breaks and we want to resume from the best solution found by the algorithm at that time. If the code is N, then a random pedigree is used as the starting solution (see the section *Input files*/Parameters for an explanation on how the software constructs the random initial solution).

Running the programme

Execution of the programme can be conducted by calling the corresponding executable files (Windows, Linux or Mac version). **Hint: Program will search for input files in the local folder, so they must be there if you want everything to run smoothly.**

Input files

At the beginning, programme will ask you for a name (*user_defined_input* from here) which will be common for all files' names; i.e., if the provided text is TRIAL, files names will be *TRIAL_* plus a code related with the content of that particular file (see below). Software read ASCII files in free format. Therefore, data can be separated by blanks (at least one), tabs or commas. A description of each of the required files follows.

Parameters

- [*user_defined_input_**param***]

In this file, the needed parameters for the algorithm to work have to be entered. Additionally, the sex and age of the sampled individuals are also established. The format of the file is as follows:

```
100  ! Seed for random number generator
20  20 ! No. genotyped and virtual individuals
30  30  N  ! Total no. markers, no. markers used for parentage compatibilities, all biallelic (Y) or
   not (N)
10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10
   10  10  10  10  10  10  ! No. alleles per locus. This row has to be removed if 'Y' was the option
   in the previous line
```

```
1 4  ! Age of puberty and senescence
3 3 0  ! No. allowed errors (tolerance) for compatibility of parentage, FS families and age
100 100 100  ! Weight given to those errors
100 100 100 100  ! Relative importance of fails in 1) known pa(ma)ternities; 2) known FS(HS)
   families; 3) banned parent-offspring relationship; 4) no reproductive individuals
3  ! No. virtual age classes
2 2 −9 ! No. virtual males and females with no defined age
4 6 8 ! No. virtual males and females with age 8
0 2 6 ! No. virtual males and females with age 6
3 1 4 ! No. virtual males and females with age 4
3 ! Initial no. generations to simulate
0 ! Calculate the IBS molecular coancestry (0) or import a user defined matrix (1)
0 ! Known relationships (1) or not (0)
0 ! Known FS(HS) families (1) or not (0)
0 ! Banned relationships (1) or not (0)
0 ! Some individuals never reproducing (1) or no such information (0)
1           7          1 ! First individual sex (1-male/2-female), age (in years or months
   or whatever measure) and genotype status (1-genotyped/0-not genotyped)
2            7            1
1            6            1
2            6            1
2            5            1
1            5            1
2            5            1
2            4            1
1            4            1
1            4            1
2            4            1
1            3            1
1            3            1
2            3            1
1            2            1
2            1            1
1            1            1
2            1            1
1            0            1
2            0            1
```

Hint: Notice the different meaning and effect of the values for *tolerance* and *weights* given to 'failures'. In the first case (*tolerance*), if for example 3 is the value for the allowed molecular incompatibility between parent and offspring, the software will not mark this relationship as wrong if only three loci or less show this incompatibility. In the case of *weights*, the value is the relative importance given to not fitting the restrictions. This could be also linked to the confidence on the 'quality' of each information (e.g., reliability of the ages of individuals, the genotypes or the known relationships).

Incompatibilities in the age of ancestors at the time of an individual's birth can be detected by defining the age of puberty and senescence and, thus, the range of fertile ages. Compatibilities are not only checked between parent and offspring but also for further ancestries. For example, if the minimum reproductive age is set at two years, no individual can be the grandfather of another unless it was born four years earlier (i.e., two generations of two years each).
Hint: The range of fertile ages and can be defined based on particular species physiology or accounting for the management of the population (i.e., individuals culled at a particular age). In case of doubt, it is better to 'enlarge' this range for avoiding real ancestries to be undetected.

The software provides the option to include ‘general virtual individuals’ (i.e., with no age) or to assign a specific age to each of them. The latter option may be appropriate when there is a precise knowledge of the historical size of the population during the generations that the reconstructed pedigree comprise. If no ‘virtuals’ are to be considered these rows have to be removed.
Hint: If we include virtual individuals, a line for general ones have to be always present in the file even if we assume all ‘virtuals’ with known age; in this situation the line should read *0 0 -9*. Contrarily, if all virtual individuals are of non-defined age, the number of age classes has to be set to 0.

To obtain the initial random solution, individuals in the population are arbitrarily divided into several ordered groups. Those in the first group are assumed to be founders and no parents are assigned to them. Individuals in the second group are assigned a random father and a mother (of appropriate sex) from the individuals in the first group. Then, individuals in the third group are similarly linked to the ones in the second group and so on. The initial number of groups can be related to the prior information on the most likely structure of the pedigree to be reconstructed.
Hint: To guarantee at least one male and one female are to be included in each group, a recommendation is to limit the number of generations forced in the initial solution, especially when the number of candidates is small. The number of groups assumed for the initial pedigree does not preclude the algorithm evaluating solutions with more (or fewer) generations and also overlapping ones.

Genotypes

- [*user_defined_input_**genot***]

The basic information required to perform the reconstruction are the genotypes of the tested individuals. As said before, programme reads data in free format, so there is no need of genotypes to be in columns (although it looks nicer). Alleles should be coded, separately for each locus, correlatively from 1 to the number of alleles (a companion program is supplied to recode from other kind of numeric values, e.g., no. repeats, fragment length, etc.). Therefore, the format of the genotypes’ file is:

⇒ one individual per row and two columns per marker (first allele first locus, second allele first locus, first allele second locus, second allele second locus, …)

```
  5  5  7  7 14 14  5  5  3  4  0  0  1 23  6  7  3  7  4  4  3  8
  7  8  6  8  2 14  5  5  3  4 16 19  1  9  9 12  3  5  4  4  4  8
  6  8  2  8  3  4  5  5  1  2  3 18 10 12  5 12  3  3  4  5  5  5
  3  6  7  8  4  9  5  5  1  1 18 20  1  5  8 10  7  7  4  5  3 11
  9 11  8 12  9 10  4  5  1  3  2 18  4 19  1  6  3  3  3  4  2 10
  5  5  7 12  9 14  4  5  1  3 17 19  1 10  6 12  3  3  5  9  3  8
  8 10  7 10  5 14  2  5  1  2  2 19  1 24  6  7  3 11  4  5  4  8
  5 10  4  7  5  6  5  7  1  4 12 19 10 17  6  7  3  3  7  9  3  4
  5  9  4  8  3  4  4  5  1  3  2 16 14 19  5  6  3  5  3  7  2  4
  7 12  8  8  3  4  5  7  1  1  2 15  1 16  6  7  3  7  3  7  2  3
  6  9  2  9  2  4  5  6  1  1  2  3  1  9  6  9  3  3  3  4  3 10
  5  5  8 12  3  3  4  5  1  3 17 18  4 16  7 12  3  5  4  7  3  4
  5  6  2  9  9 12  4  6  1  3  3  5  2 12  5 13  5  6  3  8  6  8
  3  8  7  8  4 14  1  5  1  2  7 20  4  9  5 12  7  7  5  9  3  5
  5  8  8 10  9 13  5  5  1  1 12 18 13 19  6 12  7 10  2  3  3  9
```

Missing values must be coded as 0 (zero).

Hint: The genotypes should be in the same order as the sex and age information of individuals appear in the **_param* file.

Coancestry matrix

- [*user_defined_ input_**coanc***] (optional file)

Text file with the coancestry matrix to be used as comparison instead of the similarity (IBS) coancestry (Nejati-Javaremi et al., 1997). Full matrix must be entered (i.e., coancestry between *i* and *j* and between *j* and *i* are to be provided). Therefore, file should have as many rows as the number of genotyped individuals with the same number of values in each.

Known relationships

- [*user_defined_ input_**known_parentage***] (optional file)

When some parent-offspring pairs are known, incorporating these known relationships helps to stablish a 'nuclear' structure around which the rest of the pedigree can be reconstructed. These relationships will be forced to be present in the final pedigree. The file has the classical structure of a pedigree with three columns representing the code of the individual, the code of its sire and the code of its dam. If no sire (dam) is known for an individual the value must be set to zero. None, only sire, only dam or both ancestors can be determined in this file. The example below shows a situation where the maternal lineage is known (which is a common case for some species):

```
1       0       0
2       0       0
3       0       2
4       0       2
5       0       4
6       0       4
7       0       4
8       0       5
9       0       5
10      0       5
11      0       5
12      0       8
13      0       11
14      0       11
15      0       5
16      0       7
17      0       14
18      0       14
19      0       16
20      0       18
```

Hint: All 'real' individuals should appear in this pedigree, whatever genotyped or not.

Known families

- [*user_defined_ input_**known_families***] (optional file)

As in the case of parent-offspring, when groups of sibs can be identified in the sampled individuals, they can be fixed when performing the reconstruction. Notice that this could happen even if we do not know the exact mother and/or father of this family (as it may be the case of eggs laid together or seeds within the same fruit). Therefore, the way of giving the information to the software is not a simple pedigree. The format of the file requires the identification of the number of sire and dam HS families, how many individuals belong to each family and the particular code of them. FS families can be defined by including some individuals in a sire HS family and a dam HS family at the same time.

```
1 ! No. of sire HS families
4 ! No. of sibs in this family
5 ! 1st individual in the family
6 ! 2nd individual in the family
7 ! 3rd individual in the family
8 ! 4th individual in the family

1 ! No. of dam HS families
5 ! No. of sibs in this family
8 ! 1st individual in the family
9 ! 2nd individual in the family
10
11
15
```

Hint: A blank line must be always present between families.

Banned parentage

- [*user_defined_ input_**banned_parentage***] (optional file)

The software allows users to define specific relationships (actually parent-offspring pairs) that must be avoided during the reconstruction, beyond the age incompatibilities (e.g., individuals not being in the birth location of another one cannot be assigned as their parents). The file contains the following elements:

```
3 ! No. of individuals with banned sires or dams
2 ! Maximum no. of incompatibilities per individual and sex of the parent
8 2 10 15 1 20 ! Code of the individual, no. banned sires, code of the sires (as many as stated before), no. banned dams, code of the dams
9 1 10 1 18 ! 2nd individual
10 1 15 1 20
```

Individuals without offspring

- [*user_defined_ input_**no_reproduction***] (optional file)

Some individuals can be marked as 'non-reproducers' and ensuring they are never assigned offspring in the pedigree, i.e., they will be located at the end of a pedigree branch. The format of this file is simply a list of the codes of those individuals.

```
3
32
37
```

Hint: It is not necessary to include in this file individuals without offspring because they are still too young (i.e., under reproductive age) as this situation will be accounted for when checking the minimum age to be parent.

Initial solution

- [*user_defined_ input_**initial***] (optional file)

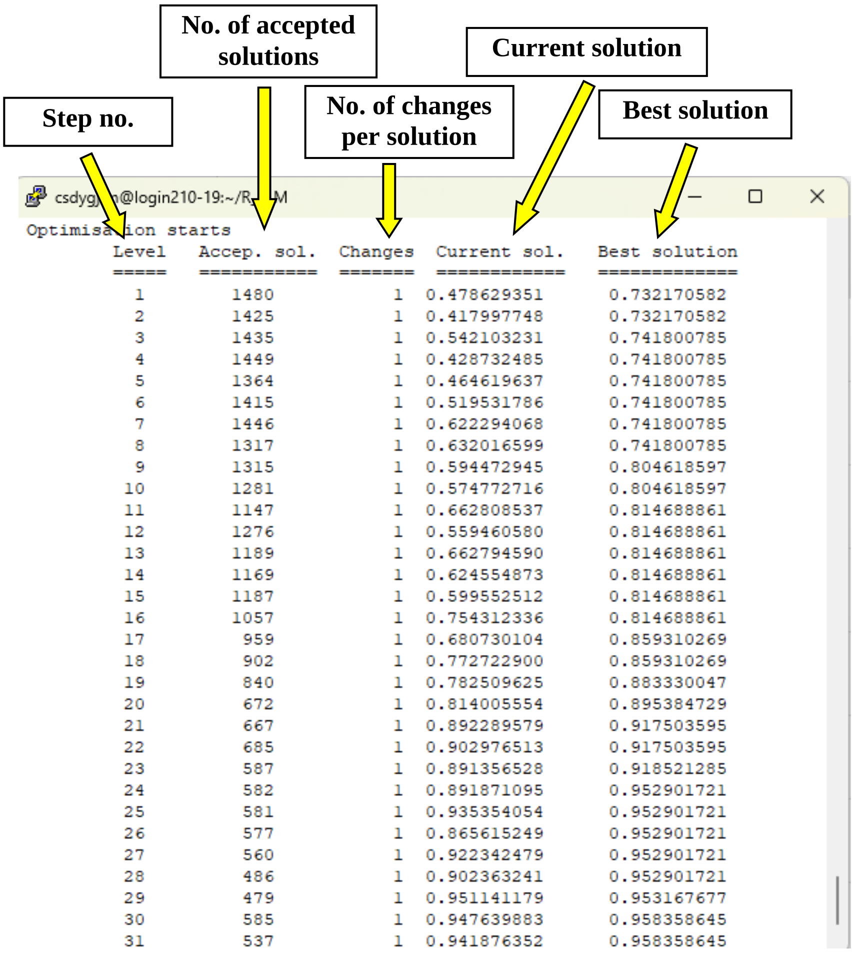
If we want the *simulated annealing* starting from a particular solution (pedigree), we have to include at the end of the ***anneal_param.txt*** file the code ‘N’, as explained before. The structure of the file is the one of a classic pedigree with columns for the individual, its sire and its dam.

```
 1     0     0
 2     0     0
 3     0     0
 4     0     0
 5     3     4
 6     3     4
 7     3     4
 8     3     5
 9     6     5
10     6     5
11     6     5
12     3     5
13    10    11
14    10    11
15    12     5
16     0    14
17    13    14
18    13    14
19    17    16
20    12    18
```

Hint: This pedigree must include all real individuals as well as the ‘virtuals’, if they have been allowed for the reconstruction.

Information on the screen

During the execution of the *simulated annealing* algorithm the programme will show some information to check if the process is running correctly and to allow for the 'fine tuning' of the control parameters. Screen will look like this


No. of accepted solutions
Current solution
Step no.
No. of changes per solution
Best solution
csdyg n@login210-19:~/R M

```
Optimisation starts
           Level    Accep. sol.   Changes   Current sol.    Best solution
           =====    ===========   =======   ============    =============
             1          1480           1  0.478629351      0.732170582
             2          1425           1  0.417997748      0.732170582
             3          1435           1  0.542103231      0.741800785
             4          1449           1  0.428732485      0.741800785
             5          1364           1  0.464619637      0.741800785
             6          1415           1  0.519531786      0.741800785
             7          1446           1  0.622294068      0.741800785
             8          1317           1  0.632016599      0.741800785
             9          1315           1  0.594472945      0.804618597
            10          1281           1  0.574772716      0.804618597
            11          1147           1  0.662808537      0.814688861
            12          1276           1  0.559460580      0.814688861
            13          1189           1  0.662794590      0.814688861
            14          1169           1  0.624554873      0.814688861
            15          1187           1  0.599552512      0.814688861
            16          1057           1  0.754312336      0.814688861
            17           959           1  0.680730104      0.859310269
            18           902           1  0.772722900      0.859310269
            19           840           1  0.782509625      0.883330047
            20           672           1  0.814005554      0.895384729
            21           667           1  0.892289579      0.917503595
            22           685           1  0.902976513      0.917503595
            23           587           1  0.891356528      0.918521285
            24           582           1  0.891871095      0.952901721
            25           581           1  0.935354054      0.952901721
            26           577           1  0.865615249      0.952901721
            27           560           1  0.922342479      0.952901721
            28           486           1  0.902363241      0.952901721
            29           479           1  0.951141179      0.953167677
            30           585           1  0.947639883      0.958358645
            31           537           1  0.941876352      0.958358645
```

For example, too rapid decrease in the number of accepted solutions (and the corresponding number of changes per new solution) may indicate a too low initial temperature, so algorithm may get stuck in the starting area. See Kirkpatrick et al. (1983) for thorough explanation of the *annealing* method and for considerations about the definition of the different parameters.

Output files

Estimated genealogy

- [*user_defined_input_**est_geneal.txt***]

This is the main result from the software, containing the reconstructed pedigree. In the beginning of the file, a summary of the characteristics of the problem (number of real/genotyped/virtual individuals, number of loci and the maximum polymorphism level, restrictions, …) can be find, as well as the time used for the system in the computation of the solution and the correlation between the comparison matrix and the pedigree coancestry calculated from the reconstructed one (the driving parameter in the optimisation process). Then, the pedigree follows in the classical way with the code of the individuals being the order in which they appeared in the input files. See an example below:

```
No. real individuals (genotyped or not)          20

 No. genotyped individuals          20

 No. virtual sires and dams


 No. initial generations           3

 No. loci          30

 Max. No. alelles/loci          10

 No. allowed mismatches (parentage, FS(HS)families, age)
           3           3           0

Calculations took    17.7031250      seconds

 Correlation between input and solution coancestry matrix
  0.924292207

           1           0           0
           2           0           0
           3           0           0
           4           0           0
           5           3           4
           6           3           4
           7           3           4
           8           3           5
           9           6           5
          10           6           5
          11           6           5
          12           3           5
          13          10          11
          14          10          11
          15          12           5
          16           0          14
          17          13          14
          18          13          14
          19          17          16
          20          12          18
```

During the execution of the optimisation a provisional pedigree is provided in each of the ‘temperatures’ of the annealing algorithm (*user_defined_input_**est_geneal_partial.txt***). This way, if the programme breaks, you will have a partial solution to start from again, not having to begin from the scratch. Moreover, if you realise that the software is not finding better solutions for many ‘temperatures’ but still many more to try, you can stop the process without losing the information.

Hint: If virtual individuals have been allowed in the reconstruction process, these are coded first and, then, the real individuals start in 1+no. virtuals. For example, if 10 ‘virtuals’ are included the first real one is number 11. The rest of individuals appear in the same order they were included in the parameter file.

Estimated coancestry

- [*user_defined_input_**est_coanc.txt***]

In this file, the pedigree coancestry matrix (in full format) corresponding to the reconstructed genealogy can be found. This can be directly used for calculations on the genetic diversity (as the expected heterozygosity) or for the management of the population (Optimal Contributions or minimum coancestry mating).

IBS coancestry

- [*user_defined_input_**IBS_coanc.txt***] (optional file)

As said above, if no particular coancestry matrix to use as the point of comparison is provided by the user, the programme computes and use the similarity or IBS matrix from the genotyped individuals. If this is the case, this matrix can be retrieved from this file.

Known limitations

The limit in the size of the problem is related with the amount of memory available in your system. No particular limits have been imposed in the number of individuals, genome length (number of loci), number of generations, etc.

By now, the software assumes individuals with separated sexes and, therefore, self-fertilisation is not possible. In future versions hermaphroditism and self-fertilisation will be included. Additionally, sex of all individuals have to be known and provided in the input file (even for the virtual individuals).

If you have any particular problem when using the program please, contact with the author in the e-mail jmj@inia.csic.es. If you discover any bug or you would like any feature to be included in future versions, your comments will be very welcome.

References

Fernández, J., Toro, M.A. 2006. A new method to estimate relatedness from molecular markers. Mol. Ecol. 15. 1657-1667. doi: 10.1111/j.1365-294X.2006.02873.x

Gómez-Romano, F., Villanueva, B., Rodríguez de Cara, M.A., Fernández, J. 2013. Maintaining genetic diversity using molecular coancestry: the effect of marker density and effective population size. Genet. Sel. Evol. 45, 38. https://doi.org/10.1186/1297-9686-45-38

Kirpatrick, S., Gelatt, C.D., Vecchi, M.P. 1983. Optimization by simulated annealing. Science 220, 671–680. doi: 10.1126/science.220.4598.67

Morales-González, E., Saura, M., Fernández, A., Fernández, J., Pong-Wong, R., Cabaleiro, S., Martínez, P., Martín-García, A., Villanueva, B. 2020. Evaluating different genomic coancestry matrices for managing genetic variability in turbot. Aquaculture 520, 734985.

Nejati-Javaremi, A., Smith, C., Gibson, J.P. 1997. Effect of total allelic relationship on accuracy of evaluation and response to selection. J. Anim. Sci. 75, 1738–1745. https://doi.org/10.2527/1997.7571738x.

Solberg, T.R., Sonesson, A.K., Woolliams, J.A., Meuwissen, T.H.E., 2008. Genomic selection using different marker types and densities. J. Anim. Sci. 86, 2447–2454. https://doi.org/10.2527/jas.2007-0010

Liability

The R_P_M software is developed and owned by INIA. Access and use of the software is free of charge. INIA (and the developers) can not be held responsible for the results coming when using the software.

How to cite this programme

Fernández, J. 2025. R_P_M: A software to reconstruct pedigrees from molecular markers. Aquaculture 602: 742370. 10.1016/j.aquaculture.2025.742370.