RePed. A tool for checking, exploring and debugging pedigrees

Jesús Ángel Baro de la Fuente (baro@arrakis.es)
Rubén Álvarez (ras@pinon.ccu.uniovi.es)
Carlos E. Carleos (carleos@pinon.ccu.uniovi.es)
David García (davidgm@eucmos.sim.ucm.es)
Hugo Lamelas (hugo@coala.uniovi.es)

Introduction

The purpose of this paper is to present a software tool that analyses large pedigrees formatted as data files with an individual-father-mother structure.

The main problems addressed were the complexity of family links and memory allocation. This large pedigree files also need an efficient and informative error test. Finally, some other utilities were added to deal with subpedigrees and genotypes.

Methods

RePed is a tool developed for checking, exploring and debugging pedigrees. RePed tasks are performed through use of a structured type consisting on an integer for the numerical identity of the animal (automatically set by the program), a string of characters for its alphanumerical identity, a real value for the inbreeding coefficient and two pointers to strutured type for the sire and dam of the animal. This pointer structure provides an interpretation of the pedigree as an oriented graph, defined as the graph formed by the union of a set of oriented binary trees.

There are two types of input files for RePed:

1. Pedigree files: These are files with at least three alphanumerical columns: the alphanumerical identity of an animal, of its sire and of its dam. A fourth column can be added containing the animal genotype.

2. Data files: These are files with the alphanumerical identity of an animal in the first column and the second column may contain the alphanumerical identity of the individual's dam.

After reading all data, non-informative animals are deleted from pedigree. An animal is considered non-informative when it is repeated or when it is a generic, unknown animal.

Animals are allocated within the structured vector by first piling up the three identity columns and simultaneously fetching each individual's sire and dam.

Error detection and correction

When all animals are stored in the structured vector, a pedigree depuration is performed to detectec the following types of errors:

1. Non consistent records: Repeated animals with different sets of parents, self-sire, or self-dam individuals.

2. Sex errors: An animal is found in both sire's column and dam one.

3. Circular pedigree: A circular pedigree error is found when an animal appears as its own ancestor.

Non consistent records and sex errors may be automatically corrected and the pedigree file may be overwritten with a corrected one. There is a choice of implemented correction policies: suspicious animals are prefixed by 0, or substituted by generic animals.

Circular pedigree errors cannot be automatically corrected. As this error must be corrected to avoid infinite loops during the execution of the program, an ascii-art subtree of the circular path is written on an error file, to guide the user for manual correction.

Each error condition (corrected or not) is documented and collected in a detailed report file.

When a pedigree has no errors, it may be analysed using genetic diversity coefficients. Implemented coefficients are imbreeding coefficient (F), effective number of founders f_e, effective number of ancestors f_a and average relatedness AR.

Inbreeding coefficient

The inbreeding coefficient may be defined as the probability of being identical-by-descent (IBD) homozygous. This may be achieved by use of the subpedigree containing all of an individual's ancestors, i.e., the oriented subtree that has this individual as vertex.

There are two methods to calculate the inbreeding coefficient:

1. First one involves searching for all possible paths between an animal's sire (i_s) and dam (i_d) passing through a common ancestor of i_s and i_d

(AN). Only paths verifying that the intersection of the set formed by animals in the path from i_s to AN path and the set formed by animals in the path from i_d to AN path is the common ancestor AN will be considered as valid ones.

Using graph theory notation:

$AN \in \left(\Gamma(i_s) \cup \Gamma^2(i_s) \cup ... \cup \Gamma^{n_s}(i_s)\right) \cap \left(\Gamma(i_d) \cup \Gamma^2(i_d) \cup ... \cup \Gamma^{n_d}(i_d)\right)$, with $\Gamma(i) = \{i_s, i_d\} = \{\text{parents of } i \text{ vertex}\} = \{\text{vertex following } i\}$.

The contribution to the inbreeding coefficient of each path takes the value $\left(\frac{1}{2}\right)^{n_j}\left(1 + F(AN(j))\right)$, where $F(AN(j))$ is the inbreeding coefficient of the j-th common ancestor of i_s and i_d and n_j is the total number of vertices in i_s to i_d path through $AN(j)$, i.e., $n_j = \sharp C_{i_s,AN(j)} \cup \sharp C_{i_d,AN(j)}$, with $C_{i,j}$ being the path between vertex i and vertex j.

Inbreeding coefficient is the sum over all common ascendents of this partial contributions.

2. The second method is based on coancestry. Letting A and B be i_s parents and C and D be i_d ones (A, B, C and D are the four grandparents of selected animal i), then the inbreeding coefficient of animal i will be $F(i) = f(i_s, i_d) = \frac{1}{4}f(A,C) + \frac{1}{4}f(A,D) + \frac{1}{4}f(B,C) + \frac{1}{4}f(B,D)$.

   This second method allows an easy implementation, but requires the construction of the coancestry matrix and memory allocation for large pedigree files.

RePed follows the first method. The strategy is as follows:

1. Firstly all common ancestors of each couple (i_s, i_d) have to be found. Let $V_{i,k}$ be the intersection of i's ancestors and k's descendents, i.e. $V_{i,k} = \left\{j/j \in \left(\Gamma^1(i) \cup \Gamma^2(i) \cup ... \cup \Gamma^{n_i}(i)\right) \cap \left(\Gamma^{-1}(k) \cup \Gamma^{-2}(k) \cup ... \cup \Gamma^{-n_k}(k)\right)\right\}$.

2. Let $V_{AN(j)} = V_{i_s,AN(j)} \cup V_{i_d,AN(j)}$ be the binary subtree of interesting vertices related to ancestor $AN(j)$. Then set $V = \bigcup_{AN(j)} V_{AN(j)}$ will be the set of vertices for obtaining the inbreeding coefficient of individual i. All superflous and non informative branches have been deleted.

3. Now, $AN(j)$'s inbreeding coefficient must be evaluated $\forall j$. If they are not calculated yet, make a recursive call to inbreeding coefficient calculation's subroutine.

4. Not all paths between i_s to i_d through $AN(j)$ are pertinent. Only paths verifying $C_{i_s,AN(j)} \cap C_{i_d,AN(j)} = AN(j)$ are required.

Finally, the inbreeding coefficient will be obtained as:

$$F(i) = \sum_{AN(j)} \sum_{C_{i_s,AN(j)} \subset \wp\left(V_{AN(j)}\right)} \sum_{C_{i_d,AN(j)} \subset \wp\left(V_{AN(j)}\right)} \left(\frac{1}{2}\right)^{\sharp C_{i_s,AN(j)} + \sharp C_{i_d,AN(j)}} \cdot \left(1 + F\left(AN(i)\right)\right)$$

Effective number of founders

Effective number of founders is a genetic diversity coefficient (Meuwissen and Luo 1992) defined as the number of equally contributing founders that would be expected to produce the same genetic diversity as in the population under study. A founder is defined as every individual with unknown parents in the population under study.

Any gene randomly sampled at any autosomal locus of a given individual has a probability $\frac{1}{2}$ of originating from its sire and a probability $\frac{1}{2}$ of originating from its dam, and, in general, a probability $\left(\frac{1}{2}\right)^n$ of originating from one animal in $\Gamma^n(i)$ (animals on the n-th previous generation). Under this hypothesis a probability of gene origin $q(i)$ can be calculated for every animal i in the pedigree.

The sum over all population founders of the probability of gene fulfils $\sum_{\{i/ifounder\}} q(i) = 1$.

The coefficient proposed by Lacy and Rochambeau for genetic variability measurement was $f_e = \frac{1}{\sum_{i \in B} q(i)^2}$ with $B = \{k/\Gamma(k) = \emptyset\}$

The algorithm used for calculating the probabilities of gene origin $\vec{q}(i)$ are detailed below:

1. Define the population under study, i.e. the group of N animals carrying the gene pool of interest.
2. Initialize a vector $\vec{q}$ with 1 for animals in the population under study and with 0 otherwise.
3. Process pedigree file counter generation wise
   - $q(k) = q(k) + \frac{1}{2}q(i)$, $\forall k \in \Gamma(i)$

4. Divide vector $\vec{q}$ by the population size N

Finally, $f_e = \frac{1}{\sum_{i \in B} q(i)^2}$ with $B = \{k / \Gamma(k) = \emptyset\}$

Effective number of ancestors

An important limitation of the effective number of founders is that it ignores potential bottlenecks in the pedigree, i.e., it does not take account of the possibility of a population formed by fullsibs of unrelated parents. In such case, f_e will be equal to 2^n if n-th previous generation were considered although genetic diversity may be explained by only two individuals.

The effective number of ancestors coefficient tries to overcome this problem looking for the minimum number of ancestors (founders or not) needed to explain the genetic diversity observed in the population.

Ancestors are selected on the basis of their expected genetic contribution. As ancestors may not be founders, the may be related, so the sum of probabilities of gene origin $\vec{q}$ over all founders may be greater than 1, and indication that some redundant information must be eliminated.

A new probability vector must be defined, $\vec{p}$, such that $p(i)$ is the contribution of individual i to genetic diversity not yet explained by the other ancestors (marginal contribution of the ancestor i).

Probability vector $\vec{p}$ could be derived from $\vec{q}$, eliminating redundancies.

The algorithm used for the calculation of the effective number of ancestors is the following:

1. Define the population under study, i.e., the group of N animals carrying the gene pool of interest
2. Assume that first $k-1$ most important ancestors are already found.
3. Delete pedigree information (sire and dam information) for the $k-1$ ancestors already found.
4. Initialize a vector $\vec{q}$ with 1 for animals in the population under study and with 0 otherwise
5. Initialize another vector $\vec{a}$ with 1 for the $k-1$ already selected ancestors and with 0 otherwise.
6. Process the pedigree counter-generation-wise
   - $q(k) = q(k) + \frac{1}{2}q(i),\ \forall k \in \Gamma(i)$
7. Process the file generation-wise

- $a(i) = a(i) + \frac{1}{2}a(k),\ \forall k \in \Gamma(i)$

8. Compute the marginal contribution $p(i)$ by substracting the contributions of the ancestors already selected: $p(i) = q(i)\big(1 - a(i)\big)$.

9. Select the k-th ancestor: $i_k / p(i_k) > p(i),\ \forall i$

10. Go to 3.

Finally divide $\vec{p}$ by N. Then the effective number of ancestors is $f_a = \frac{1}{\sum_{i \in B} p(i)^2}$ with $B = \{k / \Gamma(k) = \emptyset\}$

Average Relatedness

Average relatedness (AR) is a genetic diversity coefficient as defined by J. P. Gutierrez and J. Canon (personal communication). Average Realtedness takes account not only of coancestry but also of inbreeding, so it can be used either as a genetic variability index or as a measurement to compare inbreeding between two populations.

A vector $\vec{c}$ containing Average Relatedness coefficients is defined as $\vec{c}' = \frac{1}{n}\vec{1}'A$, with A being the coancestry matrix.

The algorithm to obtain $\vec{c}$ is shown below:

1. Initialize $\vec{c} = \vec{1}$

2. Process pedigree file counter generation wise

   - $c(k) = c(k) + \frac{1}{2}c(i),\ \forall k \in \Gamma(i)$

3. Divide $\vec{c}$ by the population size

If i is a founder, $AR(i) = c(i)$. But if i is an ancestor but not a founder, we need to relate the i-th individual with the rest of the population except its progeny. This can be done by adding $c(i)$ half of the "free" contribution of its parents:

$c(i) = \frac{1}{2}\Big(1 - \frac{1}{2}\Big(F(j) + F(k)\Big)\Big)c(i) + \frac{1}{2}\Big(c(j) + c(k)\Big)$, where $F(j)$ is the inbreeding coefficient of the $j - th$ animal.

Other utilities

RePed includes some other utilities.
The program may create subpedigrees for a given data set (read from a data file) and a master pedigree file by recursively including all ancestors of the animals on the data set to an *animal-sire-dam* structure. A pedigree created

this way is the minimal complete pedigree containing all data file individuals.

RePed can also recodify pedigree and data files, subtituting alphanumeric identities by numeric ones. This numeric identities are automaticaly set in a generation wise-order. This assingment method assures that every ancestor of an animal will have a numerical identity less than the selected animal one.

Compatibility of genotypes in data can also be studied. There is a preliminar genotype compatibility check which compares every animal's genotype with its parents' one.

After this preliminar checking, unknown genotypes can be determined using information provided by animal's progeny. If this check does not obtain animal's genotype, a second one using its parent's genotype could be applied.

References

FALCONER DS, 1989, *Introduction to Quantitative Genetics*, 3rd ed. Longman. London.

MEUWISSEN THE, LUO Z, 1992, Computing inbreeding coefficients in large populations. *Genet Sel Evol* 24:305-313.