ITEA (2000), Vol. **96A N.° 3,** 184-188

RePed. A TOOL FOR CHECKING, EXPLORING AND DEBUGGING PEDIGREES

J.A. Baro de la Fuente, R. Álvarez, C.E. Carleos, D. García, H. Lamelas

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 detect 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:

$$AN \in \left(\Gamma(i_s) \cup \Gamma^2(i_s) \cup ... \cup \Gamma^{n_s}(i_s)\right) \cap \left(\Gamma(i_d) \cup \Gamma^2(i_d) \cup ... \cup \Gamma^{n_s}(i_d)\right),$$

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., $n_j = \#C_{i_s,AN(j)} + \#C_{i_d,AN(j)}$, 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 4 grandparents of selected animal i), then the inbreeding coefficient of animal i will be

$$F(i) = f(i_s, i_d) = \tfrac{1}{4} f(A,C) + \tfrac{1}{4} f(A,D) + \tfrac{1}{4} f(B,C) + \tfrac{1}{4} f(B,D).$$

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.

$$V_{i,k} = \left\{ j / j \in \left(\Gamma^1(i) \cup \Gamma^2(i) \cup ... \cup \Gamma^{n_i}(i)\right) \cap \left(\Gamma^{-1}(k) \cup \Gamma^{-2}(k) \cup ... \cup \Gamma^{n_k}(k)\right) \right\}.$$