Theor. Appl. Genet. 49 (1977) 237-241

© by Springer-Verlag 1977

An Efficient Method of Computing the Numerator Relationship Matrix and its Inverse Matrix with Inbreeding for Large Sets of Animals

E. ter Heijden
Data Processing Division, Agriculture Canada, Ottawa (Canada)
J.P. Chesnais
I.N.R.A., Génétique Appliquée, Jouy-en-Josas (France) and
C.G. Hickman
Animal Research Institute, Research Branch, Agriculture Canada, Ottawa (Canada)

Summary. The numerator relationship matrix describes the genetic relationships between individuals of a population. Its inverse is used for the prediction of breeding values, as outlined by Henderson (1975a).

For large populations, the recursive method commonly used is difficult to apply because of the size of the relationship matrix. Recently Henderson (1975b) derived a method which allows computing the inverse of the numerator relationship matrix itself for a large number of animals, provided the population is non-inbred. The method presented here is an extension of Henderson's method to allow for inbreeding with large number of animals. It takes inbreeding into account and computes the numerator relationship matrix as well as its inverse. The method is particularly efficient in computer storage in that it allows handling of sets of animals larger than 5000 animals, and is almost as fast as the recursive method.

Introduction

The use of the numerator relationship matrix A and its inverse matrix A^{-1} in the prediction of breeding values by best linear unbiased prediction (BLUP) has been described by Henderson (1975a). He presents a method of computing A^{-1} which applies easily to large populations when inbreeding is ignored. When inbreeding is considered, the method is limited to small populations because of the amount of storage required.

The method presented in this paper computes the A and A^{-1} matrices for an inbred population one row at a time so that the amount of storage needed is considerably reduced. With modern computers, this method can handle populations in excess of 5000 animals instead of the 300 with preceding methods.

I. Computation of the numerator relationship matrix

The numerator relationship matrix A is symmetric and as defined by Wright (1922). A well-known recursive method for computing the lower part of the A matrix is to express each element a_{ij} as

$$\text{a) } a_{ij} = a_{ji} = 1/2\,(a_{pj} + a_{qj}) \quad \text{for } i > j \tag{1}$$

and

$$\text{b) } a_{ii} = 1 + \frac{a_{pq}}{2} \quad \text{for } i = j \tag{2}$$

where p and q are the parents of i. If p and/or q is unknown, the same formula is used with $a_{pj} = 0$ and/or $a_{qj} = 0$. This method involves the search of most of the coefficients previously computed in the matrix. However, each off-diagonal element of the A matrix can be described as a linear function of previously computed diagonal elements.

In general neither p nor q is equal to j in the expression of a_{ij} given by (1). Thus the recursive relation (1) can be re-applied to the terms a_{pj} and a_{qj} and the results substituted so that a_{ij} is expressed as the sum of four terms.

The terms involving unknown parents, if any, can be eliminated since their value is known to be zero. This back substitution using the recursive relation is continued on all remaining terms which are not diagonal.

A.R.I. Contribution No. 640

At any intermediate stage in this process a_{ij} has the general form:

$$a_{ij} = \frac{a_{k,l}}{2^x} + \frac{a_{m,n}}{2^y} \dots + \frac{a_{r,s}}{2^z}$$

Ultimately all terms used to express a_{ij} are diagonal elements corresponding to older animals than animal i. In consequence only diagonal values need be stored in order to build the A matrix.

This is shown clearly in the example.

Programming Strategy

For convenience computation is done a row at a time. It begins by the row referring to the oldest animal and ends with the row referring to the youngest one. The values of the diagonal elements are progessively stored in order of animal number. In each row all terms to the left of the diagonal are computed as functions of diagonal elements of previous rows, by repeated use of the preceding process. The diagonal element itself is computed using relation (2), where a_{pq} coefficient is converted to a diagonal form, and stored. Once all elements up to the diagonal have been computed, the recursive relation (1) is applied directly to the remaining elements in the rows, since the required components have been computed in the same row.

When the computation of a row is complete, the only element that needs be saved is the diagonal. This reduces the required storage from one of order N^2 for a population of size N to one that is a linear function of N.

II. Computation of an Intermediate Triangular Matrix

Henderson (1975b) derived a method for computing the inverse of the numerator relationship matrix without calculating the matrix.

If inbreeding is taken into account, the method is limited in that large matrices have to be stored. This limitation can be efficiently reduced by applying the same strategy as that described above for computing the numerator relationship matrix itself.

According to Henderson the A matrix can be written as:

$$A = LL^1$$

where L is a lower triangular matrix and L^1 the transpose of L.

Each element of the matrix L can be expressed as:

$$\text{a)} \quad l_{ij} = 1/2(l_{pj} + l_{qj}) \quad \text{for } i > j \tag{3}$$

where p and q are the parents of i and $l_{pj} = 0$ if $j > p$, $l_{qj} = 0$ if $j > p$

$$\text{b)} \quad l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^{2}} \quad \text{for } i = j \tag{4}$$

where a_{ii} is the diagonal of the A matrix for animal i, $\sum_{k=1}^{i-1} l_{ik}^{2}$ is the sum of the squares of the off-diagonal terms of L for the row i.

Each off-diagonal element l_{ij} can be expressed as the product of l_{ij} by a scalar. This is shown by using relation (3) repeatedly on all non-zero off-diagonal terms occurring in the expression of l_{ij}.

It is thus possible to compute every off-diagonal element of the matrix L when the previous diagonal values of L have been stored (see example).

Programming Strategy

For each row of the L matrix:
a) the off-diagonal terms (first i-1 terms) are computed from diagonal elements that were derived previously and saved;
b) the diagonal term is computed using the other terms of the same row and the diagonal element of the matrix A.

This means that the matrix L can be computed a row at a time, provided the diagonal of the A matrix has been saved. This process is shown in the example.

III. Computation of the Inverse of the Relationship Matrix

The solution of the L matrix leads quickly to the A^{-1} matrix. Again following Henderson's method:

Let L = TD then

$$a^{-1} = (T^{-1})^{1} \quad (D^{-1})^{2} \quad (T^{-1})$$

D is a diagonal matrix with elements the same as the diagonal elements of L.

T^{-1} is a lower triangular matrix with specific values so that in row i:

a) the diagonal is equal to 1;

b) all off-diagonal elements are null except those for the parents of i which are equal to -1/2.

Let $T^{-1} = \{t_{ij}\}$ and $(D^{-1})^2 = \{\frac{1}{si}\}$ then

$a^{-1}_{ij} = \sum_{k=m}^{N} t_{ki}t_{kj}s_k$ where m = max(i,j) and N is the total number of animals in the population.

Programming Strategy

Several strategies might be used to build the terms of the A^{-1} matrix. The following allows one:

a) to compute each term independently

b) to avoid computations involving zero terms.

A list is made for each animal that gives its number together with the identification numbers of its direct offspring (sons and/or daughters). In order to compute the term a^{-1}_{ij} the lists for animal i and for animal j are compared in the following manner. If any element k of the list for animal i also exists in the list for animal j, then the term $t_{ki}t_{kj}s_k$ is not zero and contributes to the formation of a^{-1}_{ij} (either i and j are parents of k, or one is k itself and the other a parent of k, or both are k). Further, this term can only have one of the following three values:

$s_k/4$ if $k > i > j$ since $t_{ki} = -1/2$, $t_{kj} = -1/2$

$-s_k/2$ if $k = i > j$ since $t_{ki} = 1$, $t_{kj} = -1/2$

s_k of $k = i = j$ since $t_{ki} = 1$, $t_{kj} = 1$

Each term of the list i is compared in this manner to each term of the list j. All non-zero terms occurring in these comparisons are added to give a^{-1}_{ij}. The process allows forming each term of the A^{-1} matrix independently of other terms (see example) and completes the method for computing the inverse of the numerator relationship matrix in the presence of inbreeding.

IV. Computer Programs

Computer programs are available to compute the numerator relationship matrix and its inverse by the method described in this paper. They are devised specifically to take inbreeding into account. Four programs have been written depending on the user's needs and the information available in the pedigree:

1) computation of the relationship matrix itself, and, if desired, of its inverse;

2) same as (1) with maternal grand sires instead of dams in the pedigree;

3) computation of the inverse matrix only (fast version);

4) same as (3) with maternal grand sires instead of dams in the pedigree.

These programs are almost as fast as those that compute A^{-1} without inbreeding. They are written in FORTRAN and have been tested to 5000 animals.

V. Example

Pedigree table for 7 animals

Individual	Sire	Dam
1	0*	0*
2	0*	0*
3	1	0*
4	1	2
5	3	4
6	1	4
7	5	6

* Unknown parent

1. Computation of the numerator relationship matrix

Assume rows 1 to 5 have been computed previously Elements a_{11}, a_{22}, a_{33}, a_{44}, a_{55} have been saved.

An off-diagonal element in row 6 is derived from the pedigree table by using relation (1) repeatedly, as shown for element a_{65}:

$$\begin{aligned}
a_{65} &= 1/2(a_{15}+a_{45}) \\
&= 1/4((a_{31}+a_{41})+(a_{34}+a_{44})) \\
&= 1/8((a_{11}+a_{01})+(a_{01}+a_{21})+(a_{13}+a_{23}))+1/4a_{44} \\
&= 1/8(a_{11}+a_{11})+1/16((a_{01}+a_{01})+(a_{11}+a_{01})+ \\
&\qquad (a_{12}+a_{02}))+1/4a_{44} \\
&= 1/4a_{11}+1/16a_{11}+1/32(a_{01}+a_{01})+a_{44} \\
&= 5/16a_{11}+1/4a_{44} = 9/16 = 0.5625
\end{aligned}$$

Numerator relationship matrix A

	1	2	3	4	5	6	7
1	1.00000	0.00000	0.50000	0.50000	0.50000	0.75000	0.62500
2	0.00000	1.00000	0.00000	0.50000	0.25000	0.25000	0.25000
3	0.50000	0.00000	1.00000	0.25000	0.62500	0.37500	0.50000
4	0.50000	0.50000	0.25000	1.00000	0.62500	0.75000	0.68750
5	0.50000	0.25000	0.62500	0.62500	1.12500	0.56250	0.84375
6	0.75000	0.25000	0.37500	0.75000	0.56250	1.25000	0.90625
7	0.62500	0.25000	0.50000	0.68750	0.84375	0.90625	1.28125

Triangular matrix L for computing A^{-1}

	1	2	3	4	5	6	7
1	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
2	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	0.500000	0.000000	0.866025	0.000000	0.000000	0.000000	0.000000
4	0.500000	0.500000	0.000000	0.707107	0.000000	0.000000	0.000000
5	0.500000	0.250000	0.433013	0.353553	0.707107	0.000000	0.000000
6	0.750000	0.250000	0.000000	0.353553	0.000000	0.707107	0.000000
7	0.625000	0.250000	0.216506	0.353553	0.353553	0.353553	0.637378

The diagonal element of row 6 is derived by using relation (2), then relation (1) repeatedly:

$$a_{66} = 1+1/2a_{14}$$
$$1+1/2(a_{11}+a_{21})$$
$$1+1/4a_{11}+1/8(a_{01}+a_{01})$$
$$1+1/4a_{11} = 5/4 = 1.250$$

Note that $a_{rc} = 0$ if $r = 0$ or $c = 0$.

2. Computation of the triangular matrix L

When starting row 5, the diagonal elements of the matrix A have been saved as well as elements l_{11}, l_{22}, l_{33}, l_{44} of the matrix L. The useful information can be stored in a vector: $(l_{11}, l_{22}, l_{33}, l_{44}, a_{55}, a_{66}, a_{77})$.

The computation of an off-diagonal element is for instance:

$$l_{51} = 1/2(l_{31}-l_{41})$$
$$= 1/4(l_{41}+l_{01}+l_{11}+l_{21})$$
$$= 1/2\, l_{11}+1/8(l_{01}+l_{01}) = (1/2)l_{11} = 0.5$$

The diagonal element is:

$$l_{55} = \sqrt{a_{55}-(l_{51})^2+(l_{52})^2+(l_{53})^2+(l_{54})^2} = 0.707$$

3. Computation of the inverse of the numerator relationship matrix

Progeny table

Animal	Progeny
1	3, 4, 6
2	4
3	5
4	5, 6
5	7
6	7
7	-

For instance the computation of element a^{-1}_{43} of A^{-1} would be:

list 4 = 4, 5, 6

list 3 = 3, 5

The only element common to list 4 and list 3 is 5. Then $a^{-1}_{43} = t_{54}t_{53}d_5 = 1/4\,(d_5)^2 = 0.5$ since $5 > 4 > 3$.

Inverse of the numerator relationship matrix A^{-1}

	1	2	3	4	5	6	7
1	2.333333	0.500000	-.666667	-.500000	0.000000	-1.000000	0.000000
2	0.500000	1.500000	0.000000	-1.000000	0.000000	0.000000	0.000000
3	-.666667	0.000000	1.833333	0.500000	-1.000000	0.000000	0.000000
4	-.500000	-1.000000	0.500000	3.000000	-1.000000	-1.000000	0.000000
5	0.000000	0.000000	-1.000000	-1.000000	2.615384	0.615384	-1.230767
6	-1.000000	0.000000	0.000000	-1.000000	0.615384	2.615384	-1.230767
7	0.000000	0.000000	0.000000	0.000000	-1.230767	-1.230677	2.461534

Literature

Henderson, C.R.: Use of relationships among sires to increase accuracy of sire evaluation. J. Dairy Sci. 58, 1731-1738 (1975a)

Henderson, C.R.: Rapid method for computing the inverse of a relationship matrix. J. Dairy Sci. 58, 1727-1730 (1975b)

Wright, S.: Coefficients of inbreeding and relationship. Am. Nat. 56, 330-338 (1922)

Received July 9, 1976
Communicated by H. Skjervold

Edward ter Heijden
Data Processing Division
Agriculture Canada
Ottawa, Ontario K1A-OC6 (Canada)