ZASTOSOWANIA MATEMATYKI
APPLICATIONES MATHEMATICAE
20, 1 (1988), pp. 145–151

ALGORITHM 94

ANNA BARTKOWIAK (Wrocław)

AN ALGORITHM FOR REPEATED CALCULATIONS OF THE GENERALIZED MAHALANOBIS DISTANCE

1. Procedure declaration. For a given Gramian matrix S of size $m \times m$ and a real vector $d = (d_1, d_2, \ldots, d_m)$ the procedure calculates the Mahalanobis distance D^2 defined by formula (1). Only the lower triangle of the matrix S (stored in a one-dimensional array) is needed.

Data:

first – Boolean variable with the following meaning: if *first* $\equiv$ **true**, the matrix L^{-1} (see formula (3)) is calculated, and then the value D^2; if *first* $\equiv$ **false**, the value D^2 is calculated by formula (1), using the matrix L^{-1} calculated previously;

m – dimension of the vector d and size of the matrix S, which is an $(m \times m)$-matrix;

$d[1{:}m]$ – vector of differences for which the Mahalanobis distance is calculated;

$c[1{:}\ m \times (m+1)/2]$ – array comprising (row-wise) the lower triangle of the matrix S;

eps – constant (small number, dependent on the computer accuracy), needed when calling *Maha2* with *first* $\equiv$ **true** (e.g., $\varepsilon = 10^{-9}$).

Results:

Maha2 – value of D^2;

$c[1{:}\ m \times (m+1)/2]$ – array comprising the elements of L^{-1};

ind$[1{:}\ m]$ – comprises information on linear dependences among successive rows of the matrix S:

$$ind[i] = \begin{cases} 1 & \text{the } i\text{-th row is linearly independent of the rows } 1, 2, \ldots, i-1, \\ 0 & \text{the } i\text{-th row is linearly dependent on the rows } 1, 2, \ldots, i-1; \end{cases}$$

m1 – number of linearly independent rows in the matrix S.

```
real procedure Maha2(first,m,d,c,eps,ind,m1);
  value m,eps;
  integer m,m1;
  real eps;
  Boolean first;
  integer array ind;
  array d,c;
  begin
    integer i,j,k,n,p,q,r,s,t,u;
    real x,y;
    if first then
    begin
      comment calculate L such that S=L*L';
      n:=p:=0;
      for i:=1 step 1 until m do
      begin
        q:=p+1;  ind[i]:=1;   r:=0;
        for j:=1 step 1 until i do
        begin
          x:=c[p+1];
          for k:=q step 1 until p do
          begin
            r:=r+1;  x:=x-c[k]*c[r]
          end k;
          r:=r+1;  p:=p+1;
          if i=j then
          begin
            if x<=eps then
            begin
              ind[i]:=0;  n:=n+1;  c[p]:=.0
            end x<=eps
            else c[p]:=1.0/sqrt(x)
          end i=j
          else c[p]:=x*c[r]
        end j
      end i;
      if n>0 then
      begin
        p:=k:=0;
        comment reduce the matrix L if necessary;
```

```
    for i:=1 step 1 until m do
    begin
      if ind[i]=1 then
        for j:=1 step 1 until i do if ind[j]=1 then
          begin
            p:=p+1;  c[p]:=c[k+j]
          end ind[j]=1, j;
      k:=k+i
    end i
  end n>0;
  n:=m1:=m-n;
  comment calculate the L-inverse;
  p:=r:=t:=0;
  for i:=2 step 1 until n do
  begin
    p:=p+1;  r:=r+i;  y:=c[r+1];
    for j:=2 step 1 until i do
    begin
      p:=p+1;  s:=t:=t+1;  u:=i-2;  x:=.0;
      for k:=r step -1 until p do
      begin
        x:=x-c[k]*c[s];  s:=s-u;  u:=u-1
      end k;
      c[p]:=x*y
    end j
  end i
end first
else n:=m1;
comment reduce the vector d if necessary;
if n≠m then
begin
  k:=0;
  for i:=1 step 1 until m do if ind[i]=1 then
    begin
      k:=k+1;  d[k]:=d[i]
    end ind[i]=1, i
end n≠m;
comment calculate the Mahalanobis distance using the
        reduced vector d;
k:=0;  y:=.0;
```

```
for i:=1 step 1 until n do
begin
  x:=.0;
  for j:=1 step 1 until i do x:=x+d[j]*c[k+j];
  k:=k+i;   y:=y+x*x
end i;
Maha2:=y
end Maha2
```

Remark. The values of *ml*, *c* and *ind* must not be altered between successive calls of *Maha2*.

2. Method used.

2.1. The Mahalanobis distance receives much attention in modern data analysis.

Let $x = (x_1, \ldots, x_m)$ and $y = (y_1, \ldots, y_m)$ be two points of the space $\boldsymbol{R}^m$. Let $d = (d_1, \ldots, d_m)$ be the vector of differences

$$x - y = (x_1 - y_1, x_2 - y_2, \ldots, x_m - y_m).$$

The *Mahalanobis distance* D^2 is defined by the formula

$$D^2 = dS^{-1}d^T, \tag{1}$$

where S is a Gramian matrix determining a metric in the space $\boldsymbol{R}^m$. In statistical applications S is the covariance matrix of a random variable $X = (X_1, \ldots, X_m)$. The distance D^2 is called also the *generalized distance* between x and y (see, e.g., [2], p. 35). It is often used in biological applications, also in cluster analysis.

2.2. Taking into account that S is a Gramian matrix, we can make the Cholesky decomposition

$$S = LL^T, \tag{2}$$

where L is a lower triangular matrix with nonnegative elements on the diagonal, and all elements above the diagonal equal to zero [3].

We may invert S by the formula

$$S^{-1} = (LL^T)^{-1} = (L^T)^{-1}L^{-1}. \tag{3}$$

It is easy to show that L^{-1} is also lower triangular.

It follows that the value D^2 can be calculated as

$$D^2 = d(L^T)^{-1}L^{-1}d^T = (L^{-1}d^T)^T L^{-1}d^T.$$

Defining a vector $z = (z_1, \ldots, z_m)$ as

$$z^T = L^{-1}d^T \tag{4}$$

we obtain another formula for calculating the value of D^2, namely as the product of z and its transpose:

$$D^2 = zz^T. \tag{5}$$

2.3. The method of calculating the Mahalanobis distance by formula (5) is more economic than that by formula (1) using the inverse S^{-1}. Evaluating D^2 by formula (1) we compute a quadratic form in the matrix S. Taking into account the symmetry of S we compute D^2 as

$$D^2 = \sum_{i=1}^{m} \sum_{j=1}^{m-1} d_i d_j s^{ij} + \sum_{i=1}^{m} d_i^2 s^{ii}, \tag{6}$$

where $\{s^{ij}\} = S^{-1}$. Calculating by formula (6) we need to carry out $m(m+3)$ multiplications. Evaluating D^2 by formulae (4) and (5) and taking into account that L^{-1} is lower triangular, we need to perform $m(m+3)/2$ multiplications. Therefore, repeating the calculations many times for the same matrix S and different vectors x and y, the gain in computing time may be essential.

2.4. Suppose now that the matrix $S = \{s_{ij}\}$ is not positive definite and there exists at least one row that is linearly dependent on others. In this case a generalized Mahalanobis distance is defined by the formula

$$D^2 = dS^- d^T, \tag{7}$$

where S^- is a generalized inverse of S. Some formal properties of the generalized Mahalanobis distance may be found in [1].

We proceed now as follows:

Performing the Cholesky decomposition (calculating the matrix L from (2)) we use the formulae

$$\begin{aligned} &l_{11} = (s_{11})^{1/2}, \quad l_{k1} = s_{k1}/l_{11}, \quad k = 2, \ldots, m, \\ &\left.\begin{aligned} l_{ii} &= (s_{ii} - \sum_{j=1}^{i-1} l_{ij}^2)^{1/2}, \\ l_{ki} &= (s_{ki} - \sum_{j=1}^{i-1} l_{ij} l_{kj})/l_{ii}, \end{aligned}\right\} \quad i = 2, \ldots, m, \ k = i+1, \ldots, m. \end{aligned} \tag{8}$$

We do this sequentially overwriting the elements s_{ij} by the values of l_{ij}. At the begin of the calculations we put $ml = m$. Before calculating square roots we check whether the argument of this function is greater than zero. Taking into account rounding errors arising during computations we check the inequality

$$s_{ii} - \sum_{j=1}^{i-1} l_{ij}^2 \geqslant \varepsilon, \tag{9}$$

where $\varepsilon > 0$ is a small number dependent upon the accuracy of the computer.

Suppose that inequality (9) is not satisfied for some i $(1 \leqslant i \leqslant m)$. We put then

$$l_{ii} = l_{i+1,i} = l_{i+2,i} = l_{mi} = 0.$$

Simultaneously, the element *ind*[*i*] gets the value *0* (otherwise, if (7) is satisfied, it gets the value *1*), and we diminish the value *m1* by one.

After finishing the Cholesky decomposition we reduce the matrix L removing rows and columns that have diagonal elements equal to zero. The remaining matrix is positive definite, and its size is $m1 \times m1$ (the corresponding array c comprising this matrix has dimensions $c[1{:}\ m1 \times (m1+1)/2]$).

The reduced matrix L is inverted by the usual algorithm of Martin et al. [3]. Again the elements of L are overwritten by the sequentially calculated elements of L^{-1}.

2.5. The Cholesky decomposition and the inversion of L are done only once, when calling *Maha2* with *first* ≡ **true**.

Entering the function *Maha2* with *first* ≡ **false**, we check whether $m1 = m$. If this is not the case, we reduce the vector d removing the elements d_i with the subscripts i such that $ind[i] = 0$. Next we perform the multiplications according to formulae (4) and (5), calculating the vector $z = (z_1, \ldots, z_{m1})$ and the appropriate value of D^2.

3. Certification. The results of *Maha2* were checked by comparing the values of D^2 with appropriate values obtained by calculations using the definition formula (1). To obtain S^{-1} we used the procedure *cholinversion2* from [3]. We got the same results.

4. Test example. The following examples were calculated on the ODRA 1305 computer (this computer is compatible with the ICL 1900 series of computers):

EXAMPLE 1.

(a) Data:

$$first \equiv \textbf{true}, \quad m = 5,$$

$$d[1{:}5] = [1.0 \quad 2.0 \quad 3.0 \quad 4.0 \quad 5.0],$$

$$c[1{:}15] = \begin{bmatrix} 1.0 \\ .0 & .0 \\ .0 & .0 & 4.0 \\ .0 & .0 & .0 & .0 \\ .0 & .0 & .0 & .0 & 9.0 \end{bmatrix},$$

$$eps = 10^{-9}$$

Results:

$$Maha2 = 6.0277778, \quad m1 = 3,$$

$$d[1{:}3] = [1.0 \quad 3.0 \quad 5.0],$$

$$c[1{:}6] = [1.0 \quad .0 \quad 1/2 \quad .0 \quad .0 \quad 1/3],$$

$$ind[1{:}5] = [1 \quad 0 \quad 1 \quad 0 \quad 1].$$

(b) Calling *Maha2* the second time with actual values:

$$first \equiv \mathbf{false}, \qquad m = 5,$$

$$d[1{:}5] = [6.0 \quad 7.0 \quad 8.0 \quad 9.0 \quad 10.0],$$

$c[1{:}15]$ – as obtained from the previous call of *Maha2*,
eps – arbitrary,
$ind[1{:}5]$ – as obtained from the previous call of *Maha2*.
We get the following results:

$$Maha2 = 63.1111111, \qquad d[1{:}3] = [6.0 \quad 8.0 \quad 10.0].$$

The values of c, ind and $m1$ are not changed during this call of *Maha2*.

EXAMPLE 2.
Data:

$$first \equiv \mathbf{true}, \qquad m = 2,$$

$$d[1{:}2] = [1.0 \quad 1.0],$$

$$c[1{:}3] = [4.0 \quad 5.0 \quad 6.5],$$

$$eps = 10^{-9}.$$

Results:

$$Maha2 = 0.5, \qquad d[1{:}2] \text{ – unchanged},$$

$$c[1{:}3] = [0.5 \quad -2.5 \quad 2.0],$$

$$ind[1{:}2] = [1 \quad 1], \quad m1 = 2.$$

References

[1] K. V. Mardia, *Mahalanobis distances and angles*, pp. 495–511 in: P. R. Krishnaiah (ed.), *Multivariate Analysis. IV*, North Holland, 1977.

[2] F. H. C. Marriott, *The Interpretation of Multiple Observations*, Academic Press, London–New York 1974.

[3] R. S. Martin, G. Peters and J. H. Wilkinson, *Symmetric decomposition of a positive definite matrix*, Numer. Math. 7 (1965), pp. 362–383.

INSTITUTE OF COMPUTER SCIENCE
UNIVERSITY OF WROCŁAW
51-151 WROCŁAW

Received on 1986.06.19