

TAWNY – a subroutine to update a matrix inverse

1 Introduction

Inversion of a symmetric, positive definite matrix of size $n \times n$ involves computations proportional to n^3 . At times, the matrix is build up gradually, i.e. we might have previously inverted a matrix of size $n_1 \times n_1$ and now added $n_2 = n - n_1$ new rows and columns. It is then possible to reduce computational effort by re-using results from previous computations, i.e. to update the inverse rather than starting ‘from scratch’.

A typical application, considered by Meyer et al. (2013) is the inversion of the genomic relationship matrix when new individuals are genotyped. TAWNY is a subroutine, written in FORTRAN, which implements the strategy described in this publication.

2 Source code

TAWNY is distributed as source code only, as efficient executables require compilation with libraries tuned to the specific hardware employed. It is coded as a FORTRAN module to be loaded with your main program.

There are two versions:

1. A slow, “standalone” version not utilizing any library routines. This is held in file `tawny_ind.f90`.
2. A version utilizing routines from the BLAS and LAPACK libraries, provided in file `tawny_las.f90`.

2.1 Options

Several options can be set at the beginning of `tawny_las.f90` or `tawny_ind.f90`.

```
module tawny_module

!   Version using BLAS and LAPACK routines

!   ----- select options -----
!   how to: inversion - with or without intermediate steps/files
character(len=6)                :: how_invert = 'intmed' ! 'inonly'
!   how to: cholesky factor/inverse
character(len=6)                :: how_factor = 'dpftrf' ! 'dpotrf'
!   how to: full-store or pack G11
character(len=6)                :: how_11stor = 'fulmat' ! 'packed'
!   check inverse
logical                        :: checkinv   = .true.  ! .false.
```

- a) Option `check_inv`, set to `.true.` or `.false.`, selects whether the inverse calculated is checked by multiplying it with the original matrix and evaluating the average diagonal and off-diagonal elements of the product (which should be an identity matrix). Note that computations involved can take quite some time for larger matrices.
- b) Option `how_11stor` has valid values ‘fulmat’ and ‘packed’. It determines whether the $n_1 \times n_1$ submatrix corresponding to the previously inverted rows and columns is stored as full matrix in a two-dimensional array or in packed form, i.e. the elements of the upper triangle only in a vector of length $n_1(n_1 + 1)/2$.
- c) Option `how_invert` only affects calculations when there are no previous results, i.e. $n = n_2$. A value of ‘inonly’ chooses to perform an inversion only, while ‘intmed’ causes intermediate results – to be used in the next updating step – to be written out.
- d) Option `how_factor` only has an effect `how_11stor` is ‘fulmat’. It selects the storage scheme and LAPACK routines used to calculate the Cholesky factor and inverse of

the 'new' diagonal block: 'dpotrf' and 'dpftrf' select routines DPOTRF together with DPOTRI and DPFTRF together with DPFTRI (rectangular full packed format), respectively.

2.2 File names

TAWNY uses 'standard', self-explanatory names for the five in- and output files containing the new matrix to be inverted (GNRM_New.dat) and its inverse (GNRMInv_New.dat), the new (CholInv_New.dat), and old (CholInv_Old.dat) inverse Cholesky factors and the old inverse (GNRMInv_Old.dat). These can be easily modified by editing the respective lines.

```
! ----- set "standard" file names to be used -----
! standard file names
! character(len=15), parameter      :: ginvold = 'GNRMInv_Old.dat', &
&                                  ll11inv  = 'CholInv_Old.dat',   &
&                                  ginvnew  = 'GNRMInv_New.dat',   &
&                                  gnrnew   = 'GNRM_New.dat',      &
&                                  mm11inv  = 'CholInv_New.dat'
! ----- end of user modifications -----
```

2.3 Input and output

TAWNY uses a call to a subroutine named `rdwr_matrix` to acquire the matrix to be inverted (and any previous results) and write out results. It has 4 parameters:

1. `gmode` is a character variable of length 5 and should be equal to 'read' or 'write'.
2. `fname` is a character variable of length 15 containing the name of the file to be read or written to.
3. `nrow` is an integer variable giving the number of rows and columns in the matrix to be read or written out.
4. `cc` is the name of the double precision vector in which stores the upper triangle of the matrix.

```
! =====
! subroutine rdwr_matrix( gmode, fname, nrow, cc)
! =====

implicit none
character(len=5), intent(in)      :: gmode
character(len=15), intent(in)     :: fname
integer, intent(in)              :: nrow
real(8), dimension(nrow*(nrow+1)/2), intent(inout) :: cc
```

This routine is not part of TAWNY, i.e. it is expected that this routine is supplied by the user, as the preferred formats may differ greatly. For large matrices, input and output in binary format is advantageous. An example for `rdwr_matrix` is given in the test program supplied.

2.4 Test program

An example for the use of TAWNY is given in file `test_tawny.f90`. For the version using BLAS and LAPACK routines, this needs to be compiled loading the appropriate libraries. Example compile options for a Linux operating system are provided in make files `make_las` (using `gfortran` together with system libraries) and `make_mkl` (using `ifort` together with Intel MKL libraries).

A toy example ($n_1 = 8$, $n_2 = 2$) is provided in files `GNRM_New.dat`, `CholInv_Old.dat` and `GNRMInv_Old.dat`. For portability, these are formatted files and `test_tawny.f90`

is set up accordingly.

2.5 Download

Tawny if available for download at:

<http://didgeridoo.une.edu.au/km/tawny.php>

3 Conditions of use

Please credit the use of TAWNY in any publications using the reference to Meyer et al. (2013) given below.

TAWNY is available to the scientific community free of charge. Permission is granted to re-distribute it complete and unmodified, but not for profit. Modifications for personal use are allowed but should be clearly marked.

Disclaimer: While every effort has been made to ensure that TAWNY does what it claims to do, there is absolutely no guarantee that results are correct. Use is entirely at your own risk !

References

Meyer K., Tier B., Graser H.U. Technical note: Updating the inverse of the genomic relationship matrix. J. Anim. Sci. 00 (2013) 000.



Tawny Frogmouth