Assignment 3: Dense Linear Algebra Libraries

Lars Karlsson (assistant)
Bo Kågström
Mikael Rännar
Robert Granat

Lab session: 2009-04-23 (8.15-10.00)
Due date: 2009-04-28 (17.00)

1 Mathematical Background

Consider the following overdetermined system of equations with multiple right-hand sides:

$$AX = B,$$

where $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times k}$, and consequently $X \in \mathbb{R}^{n\times k}$. Since the system is overdetermined we have $m > n$ and A has full rank.

Such systems occur frequently in model fitting and generally they have no solution so we have to settle with an approximation. A common choice is the *linear least squares approximation* which is the X that minimizes the residual norms

$$||Ax_{\bullet j} - b_{\bullet j}||_2$$

for each right-hand side $b_{\bullet j}$. Note that the k right-hand sides define k independent linear least squares problems. Minimizing the 2-norm of each residual is equivalent to minimizing the Frobenius norm of the matrix residual

$$||AX - B||_F.$$

The Frobenius norm of an $m \times n$ matrix A is defined as

$$||A||_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2}.$$

We begin by finding a QR factorization of A.

$$A = QR = \begin{bmatrix} Q_A & Q_\perp \end{bmatrix} \begin{bmatrix} R_A \\ 0 \end{bmatrix}.$$

Since the Frobenius norm is invariant under orthogonal transformations we get

$$||AX - B||_F = || \begin{bmatrix} R \\ 0 \end{bmatrix} X - Q^T B||_F = || \begin{bmatrix} R_A X - Q_A^T B \\ -Q_\perp^T B \end{bmatrix} ||_F.$$

The second block row is constant and its norm gives the minimum residual. The first block row can be made zero by solving for X in the linear system

$$R_A X = Q_A^T B.$$

This system has a unique solution since R_A is nonsingular due to the full rank condition on A. The $n \times n$ matrix R_A is upper triangular which means that we can solve the system using back substitution.

2 Assignment

Your task is to fill out the missing details in a skeleton subroutine in Fortran that finds the least squares approximation to an overdetermined system of linear equations using the method detailed in the previous section.

These are the major steps:

- Initialize BLACS, read and broadcast input parameters, generate matrices.
- Copy A and B so you can verify your solution later.
- Compute a QR factorization of A. This will overwrite A with

$$\begin{bmatrix} R \\ 0 \end{bmatrix}.$$

  However, it will use the positions of the zeroes to store Q in factored form. For this purpose, we also need an auxiliary vector τ (`TAU`) to store some multipliers that do not fit inside A.
- Compute $B \leftarrow Q^T B$ using the factored form of Q that is stored in A.
- Solve the linear system

$$R_A X = Q_A^T B$$

  using back substitution.
- Compute the residual

$$AX - B.$$

  Now you need your copies of A and B since they have been overwritten by now.
- Compute the Frobenius norm of the residual. For square systems ($m = n$) this should vanish (say 10^{-10}) but for $m > n$ you should expect a moderate (around tens and hundreds) norm.

In the skeleton file `lab3-skeleton.f90` there are places marked with `todo` which precedes a couple of deleted lines that your task is to reconstruct. There are in all cases just a handful up to a dozen of lines to add in each place and the context should be helpful. Besides the main routine there are two subroutines at the bottom of the source file which contain some missing pieces too. Make sure that you understand what everything in the program does and what every parameter to every function represents.

3 Guide to Provided Files

Download the tarfile `assignment3-skeleton.tgz` from the course web. This file contains the following tree structure:

```
assignment3-skeleton/test/inp.dat
assignment3-skeleton/test/submit
assignment3-skeleton/Makefile
assignment3-skeleton/assignment3-skeleton.f90
assignment3-skeleton/drand_.c
```

The `drand_.c` file contains a wrapper for a random number generator. The details are not important. The `Makefile` builds and links an executable called `assignment3` from `assignment3.f90` and `drand_.c`, so you must first copy the skeleton file into `assignment3.f90`. You need to modify the `submit` file and in particular the account and the number of processes. You are now ready to solve the assignment by suitably modifying `assignment3.f90` based on the instructions in the source and this short note.