Scientific Computing for DPhil Students I
Assignment 2

Due at lecture at 11:00 on Tues. 6 Nov. 2018. This is the second of four assignments this term.

Here is our policy on collaboration. By all means ask friends for help on general questions like "How do I compute a condition number in MATLAB?" or "what's a good reference on conjugate gradients?" Do not, however, discuss specifics of these four problems.

Make your MATLAB codes clean, elegant, and short. Remember to turn in listings.

In the first three problems we are going to consider the Poisson equation $-\Delta u = f(x, y, z)$ on the cube $[0, 1]^3$ with boundary condition $u = 0$, where $-\Delta u = -\partial^2 u/\partial x^2 - \partial^2 u/\partial y^2 - \partial^2 u/\partial z^2$. To approximate $-\Delta$ numerically, we pick an integer $n \geq 2$, set $h = n^{-1}$, and consider the $(n-1)\times(n-1)\times(n-1)$ grid of regularly spaced points (ih, jh, kh) in the cube with $1 \leq i, j, k \leq n-1$. We then use the standard 7-point centred difference approximation

$$(Au)_{i,j,k} = n^2(6u_{i,j,k} - u_{i+1,j,k} - u_{i-1,j,k} - u_{i,j+1,k} - u_{i,j-1,k} - u_{i,j,k+1} - u_{i,j,k-1}).$$

We can generate this matrix with the following slight variation of the code given in Problem 3 of Assignment 1:

```
D = @(n) sparse(toeplitz([2 -1 zeros(1,n-3)]));
I = @(n) speye(n-1);
A = @(n) kron(I(n),kron(I(n),D(n))) + kron(I(n),kron(D(n),I(n))) + ...
                   kron(D(n),kron(I(n),I(n)));
```

1. *Condition numbers.* Determine the condition number $\kappa(A)$ for $n = 3, 4, \ldots, 12$ by dense matrix methods using `eig(full(A(n)))`. Print a table of your results and also display them in a log-log plot (MATLAB: `loglog`). If $\kappa(A) \sim Cn^\alpha$ as $n \to \infty$, what do you think C and α are?

2. *Solution by direct methods.* Taking b for simplicity to be the vector of all ones, solve $Ax = b$ for x using the backslash operator in MATLAB for $n = 5, 6, \ldots, 30$ (or higher) and measure the computer time t for each solution with `tic` and `toc`. Display your results on a log-log plot (there's no need for a printed table). If $t \sim Cn^\alpha$, what is α? What would α be if "\" used dense Gaussian elimination without taking any advantage of sparsity?

3. *Solution by CG.* Solve $Ax = b$ to approximately 10 digits of accuracy for $n = 5, 6, \ldots, 40$ (or higher) by the CG iteration without a preconditioner using MATLAB's `pcg` or a modification of the code `cg` from our course web site, and again make a log-log plot of t against n. What is the exponent α now? How much of α represents work per step of the CG iteration, and how much represents the number of steps needed for convergence? How can you explain the latter in view of the results of Problem 1? For $n = 50$, what do you estimate the times would be for the direct method and for unpreconditioned CG? What about $n = 100$?

4. This last problem is unrelated to the others. Let A be the 100,000 × 100,000 matrix whose main diagonal consists of the prime numbers $2, 3, 5, \ldots, 1{,}299{,}709$ and whose other entries are all 0 except that $a_{ij} = 1$ for any i and j with $|i - j| = 2^k$ for some $k \geq 0$. If $b = (1, 2, 3, \ldots, 100000)^T$ and $Ax = b$, what is x_{10000}? (You may find the MATLAB commands `primes` and `spdiags` useful.) Explain how you've solved the problem and show how much computer time it takes. For full marks, your solution should be a speedy one.