Lecture 3: Sci. Comp. for DPhil Students

Nick Trefethen, Tuesday 22.10.19

Today

- I.6 Preconditioned CG
- I.7 Examples of preconditioners

Good reading for this lecture: Trefethen & Bau chaps 38 and 40

Handouts

- solution sheet for Assmt. 1
- `m4_indices.m` - playing with Matlab indices
- p. 1 of Meijerink-van der Vorst 1977 paper
- `prcg.m` and `m05_pcg.m` - preconditioned CG

Announcements

- Assignment 1 is due now.
- Assignment 2 will be handed out Thursday.
- Around 60 signed the sheets on Friday, about 2/3 of whom say they plan to take the course for credit. Did we miss any?
  If you'd prefer not to have your name on a public list, please email me.

Here's a little Matlab to start with: indices, "find", etc. The purpose of this and other Matlab instruction in this course, let me remind you, is not just to teach you Matlab per se. Equally, it's to raise your expectations of the high level at which one can and should operate computationally.

`m04_indices.m`

I.6 Preconditioned CG

Last lecture we discussed the CG iteration for SPD problems $Ax = b$. For more information, see half the books on our reading list.

We related the error of CG at step n to the question of how small a degree n polynomial p_n normalised by $p_n(0) = 1$ can be at the eigenvalues of A.

Now to finish let story: define the the condition number of an SPD matrix A (the definition is different for non-SPD matrices) like this:

$$\kappa(A) = \frac{\lambda_{max}(A)}{\lambda_{min}(A)}.$$

Our theorems have this

Famous Corollary

$$\frac{\|e_n\|_A}{\|e_0\|_A} \le 2\left(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\right)^n \approx 2\exp\left(\frac{-2n}{\sqrt{\kappa(A)}}\right).$$

Since $e^{-2} \approx 0.14$, CG requires a little more than $\sqrt{\kappa(A)}$ steps per digit.

To illustrate, let's get back to the second half of `m3_CGconvergence.m`.

`[m03_CGconvergence.m]`

1000×1000 SPD matrix with condition number 2.8e6: no useful convergence.

After adding $100I$, the condition number shrinks to around 40, with square root around 6.3, and we get convergence to 10 digits in 85 iterations.

What a neat trick! We can speed up solutions by shifting by a multiple of I! This is useless, however, for there's no way to relate the solution of $(A + cI)x = b$ to that of $Ax = b$.

Preconditioning is a related but multiplicative idea. For example, a **left preconditioner** is a nonsingular matrix M such that

$$M^{-1}Ax = M^{-1}b \qquad (\Leftrightarrow Ax = b)$$

is better behaved (e.g. better conditioned) than $Ax = b$, yet M^{-1} can be computed fast. We don't really mean that the inverse is computed, but that systems $My = c$ are solved fast.

If A is SPD and $M = C^TC$ for some C, then the symmetry can be preserved by preconditioning via the equation

$$[C^{-1}AC^{-T}]C^Tx = C^{-1}b \qquad (\Leftrightarrow Ax = b) \qquad (*)$$

The matrix on the left is SPD, so we can use CG to solve $(*)$.

Many people claim credit for the idea of preconditioning; I won't take sides. Preconditioning became famous in the 1970s. A key paper was that of Meijerink and van der Vorst in 1977, which introduced preconditioners based on **incomplete factorization**. As of today it has 2322 hits at Google Scholar.

`[p 1 of Meijerink & van der Vorst]`

Here's a code to illustrate preconditioned CG. What's going on is essentially $(*)$, but written in a way so that we don't use C explicitly.

`[prcg.m , m05_pcg.m]`

I.7 Survey of preconditioners

See chapter 40 of Trefethen & Bau for general comments, and the other books on the reading list for details.

1. Diagonal scaling = "Jacobi"
2. Incomplete factorisation (In MATLAB see LUINC and CHOLINC)
3. Coarse-grid approximation (-> multigrid)
4. Local approximation (omitting long-range interactions)
5. Block preconditioners and domain decomposition
6. Low-order discretization (e.g. approximate spectral by FEM)
7. Constant-coefficient or symmetric approximation
8. Splitting of multi-term operator
9. Dimensional splitting or ADI
10. One step of a classical iteration (Gauss-Seidel, SOR, SSOR. . .)
11. Periodic approximation to nonperiodic problem (Toeplitz → circulant)
12. Unstable direct method (iteration may circumvent instability)
13. Polynomial preconditioners
14. Sparse approximate inverses
15. Low-rank approximations

At Oxford we have a world expert on preconditioners: Prof. Andy Wathen.