Lecture 8

Error Bounds and Residuals

In addition to being a reliable indicator of near singularity, the condition number also provides a quantitative bound for the error in the compounded solution to a linear system, as we will now see.

Let $\boldsymbol{x}$ be the solution to $\boldsymbol{Ax = b}$, and let $\hat{\boldsymbol{x}}$ be the solution to $\boldsymbol{A}\hat{\boldsymbol{x}} = \boldsymbol{b} + \Delta\boldsymbol{b}$ with a perturbed right hand side. If $\Delta\boldsymbol{x} = \hat{\boldsymbol{x}} - \boldsymbol{x}$, then

$$\boldsymbol{b} + \Delta\boldsymbol{b} = \boldsymbol{A}\hat{\boldsymbol{x}} = \boldsymbol{A}(\boldsymbol{x} + \Delta\boldsymbol{x}) = \boldsymbol{Ax} + \boldsymbol{A}\Delta\boldsymbol{x}$$

$$\boldsymbol{i.e.}\Delta\boldsymbol{b} = \boldsymbol{A}\Delta\boldsymbol{x} \Rightarrow \Delta\boldsymbol{x} = \boldsymbol{A}^{-1}\Delta\boldsymbol{b} \le \left\|\boldsymbol{A}^{-1}\right\| \cdot \left\|\Delta\boldsymbol{b}\right\|$$

Which leads to the bound

$$\frac{\left\|\Delta\boldsymbol{x}\right\|}{\left\|\boldsymbol{x}\right\|} \le \boldsymbol{cond}(\boldsymbol{A}) \frac{\left\|\Delta\boldsymbol{b}\right\|}{\left\|\boldsymbol{b}\right\|}$$

for possible relative change in solution due to relative change in RSH ***b.***

Thus, the condition number of the matrix is an "amplification factor" that bounds the maximum relative change in the solution due to a given relative change in the RHS vector.

A similar result holds for relative change in the entries of the matrix ***A.*** If $\boldsymbol{Ax = b}$ and $(\boldsymbol{A} + \boldsymbol{E})\hat{\boldsymbol{x}} = \boldsymbol{b}$, then

$$\Delta\boldsymbol{x} = \hat{\boldsymbol{x}} - \boldsymbol{x} = \boldsymbol{A}^{-1}(\boldsymbol{Ax} - \boldsymbol{b}) = -\boldsymbol{A}^{-1}\boldsymbol{E}\hat{\boldsymbol{x}}$$

Taking norm, we get

$\left\|\Delta\boldsymbol{x}\right\| \le \left\|\boldsymbol{A}^{-1}\right\| \cdot \left\|\boldsymbol{E}\right\| \cdot \left\|\hat{\boldsymbol{x}}\right\|$ which gives the bound $\dfrac{\left\|\Delta\boldsymbol{x}\right\|}{\left\|\boldsymbol{x}\right\|} \le \boldsymbol{cond}(\boldsymbol{A}) \dfrac{\left\|\boldsymbol{E}\right\|}{\left\|\boldsymbol{A}\right\|}$

Result: the condition number times the relative change in the problem data bound the relative change in the solution.

One simple way of interpreting the results is that the computed solution loses about $\boldsymbol{log_{10}(cond(A))}$ decimal digits of accuracy relative to the accuracy of the input. The example in the previous lecture, for instance has a condition number greater than 10^4, so we would expect no correct digits in the solution to a linear system with the matrix unless the input data are accurate to more than four decimal digits and the solution is compared using arithmetic with more than four decimal digits of precision.

As a quantitative measure of sensitivity, the matrix condition number plays the same role for the problem of solving linear systems – and yields the same type of relationship between forward and backward error – as the general notion of condition number.

Note: The matrix condition number is never less than 1

Residuals

One way to verify a solution to an equation is to substitute it into the equation and see how closely left and right sides match.

Residual vector of approximate solution $\hat{\boldsymbol{x}}$ to linear system $\boldsymbol{Ax = b}$ defined by

$$\boldsymbol{r = b - A\hat{x}}$$

In theory, if $\boldsymbol{A}$ is nonsingular, then $\|\Delta \boldsymbol{x}\| = \|\hat{\boldsymbol{x}} - \boldsymbol{x}\| = \boldsymbol{o}$, if and only if $\|\boldsymbol{r}\| = \boldsymbol{o}$. In practice, however these quantities are not necessarily small simultaneously.

Note that if the equation $\boldsymbol{Ax = b}$ is multiplied by an arbitrary non-zero constant, then the solution is unaffected, but the residual is multiplied by same factor. Thus, the residual can be made arbitrarily large or small, depending on the scaling of the problem, and hence size of the residual is meaning less, unless it is considered relative to the size of the problem data and the solution. Thus the relative residual for the approximated solution $\hat{\boldsymbol{x}}$ is defined to be $\|\boldsymbol{r}\| / (\|\boldsymbol{A}\| \cdot \|\hat{\boldsymbol{x}}\|)$.

To relate the error to the residual, we observe that

$$\|\Delta \boldsymbol{x}\| = \|\hat{\boldsymbol{x}} - \boldsymbol{x}\| = \|\boldsymbol{A}^{-1}(\boldsymbol{A}\hat{\boldsymbol{x}} - \boldsymbol{b})\| = \|-\boldsymbol{A}^{-1}\boldsymbol{r}\| \le \|\boldsymbol{A}^{-1}\| \cdot \|\boldsymbol{r}\|$$

Dividing both sides by $\|\hat{\boldsymbol{x}}\|$ and using the definition of ***cond(A)***, we then have

$$\frac{\|\Delta \boldsymbol{x}\|}{\|\hat{\boldsymbol{x}}\|} \le \boldsymbol{cond}(\boldsymbol{A}) \frac{\|\boldsymbol{r}\|}{\|\boldsymbol{A}\| \cdot \|\hat{\boldsymbol{x}}\|}$$

Thus, a small relative residual implies a small relative error in the solution when, and only when, $\boldsymbol{A}$ is well conditioned.

To see the implications of large residual, on the other hand, if computed solution $\hat{\boldsymbol{x}}$ exactly satisfies

$$(\boldsymbol{A} + \boldsymbol{E})\hat{\boldsymbol{x}} = \boldsymbol{b}$$

Then

$$\|\boldsymbol{r}\| = \|\boldsymbol{b} - \boldsymbol{A}\hat{\boldsymbol{x}}\| = \|\boldsymbol{E}\hat{\boldsymbol{x}}\| \le \|\boldsymbol{E}\| \cdot \|\hat{\boldsymbol{x}}\|$$

Which gives

$$\frac{\|\boldsymbol{r}\|}{\|\boldsymbol{A}\| \cdot \|\hat{\boldsymbol{x}}\|} \le \frac{\|\boldsymbol{E}\|}{\|\boldsymbol{A}\|}$$

So large relative residual implies large backward error in the matrix and algorithm used to compute solution is unstable. Another way of saying this is that a stable algorithm will invariably produce a solution with small relative residual, irrespective of the conditioning of the problem, and hence a small residual by itself, sheds little light on the quality of the approximate solution.

Example: Small residual

$$\boldsymbol{A}\boldsymbol{x} = \begin{bmatrix} 0.913 & 0.659 \\ 0.457 & 0.330 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} = \begin{bmatrix} 0.245 \\ 0.127 \end{bmatrix} = \boldsymbol{b}$$

Consider two approximate solutions

$$\hat{\boldsymbol{x}}_1 = \begin{bmatrix} -0.0827 \\ 0.5 \end{bmatrix} \text{ and } \hat{\boldsymbol{x}}_2 = \begin{bmatrix} 0.999 \\ -1.001 \end{bmatrix}$$

The norms of their respective residuals are

$$\|\boldsymbol{r}_1\|_1 = 2.1 \times 10^{-4} \text{ and } \|\boldsymbol{r}_2\|_1 = 2.4 \times 10^{-2}$$

so which is the better solution?

We are tempted to say $\hat{\boldsymbol{x}}_1$ because of its much small residual. But the exact solution to this system is $\boldsymbol{x} = [1,-1]^T$, as is easily confirmed, so $\hat{\boldsymbol{x}}_2$ is actually much more accurate than $\hat{\boldsymbol{x}}_1$. The reason for this surprising behavior is that the matrix A is ill-conditioned as we saw previously, and because of its large condition number, a small residual does not imply a small error in the solution.

Iterative refinement

Given approximate solution $\boldsymbol{x}_0$ to linear system $\boldsymbol{Ax} = \boldsymbol{b}$, compute residual

$$\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$$

Now solve linear system $\boldsymbol{A}\boldsymbol{z}_0 = \boldsymbol{r}_0$ and take $\boldsymbol{x}_1 = \boldsymbol{x}_0 + \boldsymbol{z}_0$ as new and "better" approximate solution, since

$$\boldsymbol{A}\boldsymbol{x}_1 = \boldsymbol{A}(\boldsymbol{x}_0 + \boldsymbol{z}_0) = \boldsymbol{A}\boldsymbol{x}_0 + \boldsymbol{A}\boldsymbol{z}_0 = (\boldsymbol{b} - \boldsymbol{r}_0) + \boldsymbol{r}_0 = \boldsymbol{b}$$

Process can be repeated to refine solution successively until convergence, potentially producing solution accurate to full machine precision**.**

Iterative refinement requires double storage, since both original matrix and LU factorization required

Due to cancellation, residual usually must be computed with higher precision for iterative refinement to produce meaningful improvement.

For these reasons, iterative improvement often impractical to use routinely, but can still be useful in some circumstances.