Math 128A, Summer 2019

Lecture 29, 8/12/2019

Plans for today:
- Matrix Norms
- Iterative Improvement

1 Floating Point Error: $Ax = b$

Suppose we want to solve the linear system, $Ax = b$. We don't know if this is solvable, but we have technology to try. The first step is to try to factorize:

(1) $PA = LU$
(2) Use the above permutation to solve $Ax = b$. Let $Pb = PAx = c$. Permute the entries of b
(3) Now let $Ly = c$
(4) Let $Ux = y$.

There are quite a few steps here; however, we sometimes cannot just 'divide' by A. Then $c = LUx = Pax$. The advantage here is that these steps are simple and routine, and a computer can do it.

For example, consider $Ly = c$ for (unit)-lower-triangular L:

$$\begin{bmatrix} l_{11} & & & 0 \\ l_{21} & l_{22} & & 0 \\ \vdots & & \ddots & \vdots \\ l_{n1} & & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} c_1 \\ \\ \vdots \\ c_n \end{bmatrix}$$

Solving this lower-triangular system gives:

$$\begin{aligned} y_1 &= c_1/l_{11} \\ y_2 &= (c_2 - l_{21}y_1)/l_{22} \\ &\vdots \\ y_n &= (c_n - l_{n1}y_1 - l_{n2}y_2 - \cdots - l_{n,n-1}y_{n-1})/l_{nn} \end{aligned}$$

1.1 Backwards Error Analysis: Floating Point Arithmetic

Of course, we get the exact result (from operations), correctly rounded **at each step of operations**. However, errors do propagate down the system. We write in terms of each slightly perturbed data values:

$$\begin{aligned} \hat{y}_1 = fl(y_1) &= (c_1/l_{11})(1+\delta_1), \qquad |\delta| \le \frac{\varepsilon}{2} \\ fl(y_2) = \hat{y}_2 &= \left((c_2 - l_{21}\hat{y}_1(1+\delta_1))(1+\delta_2)/l_{22}\right)(1+\delta_3) \\ &= \left[c_2(1+\delta_3)(1+\delta_2) - \underbrace{l_{21}(1+\delta_1)(1+\delta_2)(1+\delta_3)}_{\text{blame this for error}}\hat{y}_1\right]/l_{22} \\ &= \hat{c}_2 - \hat{l}_{21}\hat{y}_1/l_{22}, \end{aligned}$$

where we strategically let the numerators absorb the errors, and we leave l_{22} exact.
So we can write:

$$\hat{y}_n = \left(\hat{c}_n - (\hat{l}_n\hat{y}_1 + \cdots + \hat{l}_{n,n-1}\hat{y}_{n-1})\right)/l_{nn}$$

How close are these terms? We can say that $\hat{c}_n$ participates in each of the subtractions, so

$$|\hat{c}_n - c_n| \leq \varepsilon|c_n|,$$

and we conclude:

$$\left|\hat{l}_{ij} - l_{ij}\right| \leq n\varepsilon|l_{ij}|$$

In other words, in terms of matrices and vectors, for $Ly = c$ and $\hat{L}\hat{y} = \hat{c}$, we say that for each value within vector c and matrix L, we have, :

$$|\hat{c} - c| \leq \varepsilon|c|, \qquad |\hat{L} - L| \leq n\varepsilon|L|,$$

so $L_{ij} = 0 \implies \hat{L}_{ij} = 0$. This gives us our desired **backwards error analysis**.

We've shown that the computed solution is the exact solution to a slightly perturbed matrix $\hat{L}$. The backwards error is usually enough, to say that we solved a problem that is close to the original problem. However, we may need to explain (to our boss) why things went wrong, and we'll need forward error analysis. Strain says that backwards error analysis implies forward error analysis.

1.2 Forward Error Analysis

$$\begin{aligned} Ly = \hat{L}\hat{y} &= c - \hat{c} \\ Ly - L\hat{y} + L\hat{y} - \hat{L}\hat{y} &= c - \hat{c} \\ L(y - \hat{y}) + (L - \hat{L})y &= c - \hat{c} \\ L(y - \hat{y}) &= c - \hat{c} - (L - \hat{L})\hat{y} \end{aligned}$$

On the right hand side, we have the perturbation minus the perturbation in the matrix, applied to the computed solution. Now if we are given a reasonable problem to solve, then L will be invertible, so :

$$y - \hat{y} = L^{-1}$$

This is an exact formula and contains some inverses, so we want to have some generalized bound (inequality):

$$|y - \hat{y}| \leq |L^{-1}(c - \hat{c})| + |L^{-1}(L - \hat{L})\hat{y}|$$

This is more quantitative than saying we solved almost the right problem. To get even more quantitative, we say:

$$\begin{aligned} |(Ax)_i| &\leq \sum_j |A_{ij}x_j| \\ &\leq \sum_j |A_{ij}||x_j| \\ &= (|A||x|)_i \end{aligned}$$

Let $|L^{-1}|$ to be the resulting matrix by replacing all elements by each of their absolute values.
So we can say, with respect to the first inequality:

$$\begin{aligned}|y-\hat{y}| &\leq |L^{-1}(c-\hat{c})| + |L^{-1}(L-\hat{L})\hat{y}| \\ &\leq \varepsilon|L^{-1}||c| + n\varepsilon|L^{-1}||L||\hat{y}| \\ &\leq \varepsilon \underbrace{|L^{-1}||L|}|y| + n\varepsilon \underbrace{|L^{-1}||L|}|\hat{y}| \\ &\leq \boxed{(n+1)\varepsilon|L^{-1}||L||y| + n\varepsilon|L^{-1}||L||y-\hat{y}|}\end{aligned}$$

where $|c| = |Ly| \leq |L||y|$, so we tack on the penultimate simplification. It's important to notice that $|L^{-1}||L| > I$ (always larger than the identity matrix). To get the last inequality, we try writing: $\hat{y} := y - (y - \hat{y})$.
Another way to write this final bound is:

$$\left(I - n\varepsilon|L^{-1}||L|\right)|y-\hat{y}| \leq (n+1)\varepsilon|L^{-1}||L||y|,$$

and we want to multiply by the inverse. However, will multiplying by a matrix preserve monotonicity of the inequality chain? We may flip some (or all) of the inequalities. One way to get around this problem is adding all these inequalities together and multiply across by a positive value. We conclude that
Recall from lecture 28, we worked out the geometric series of matrices. We found:

$$(I-A)^{-1} = I + A + A^2 + \cdots,$$

if $|\lambda_j(A)| < 1$ for all $j \in \mathtt{1:n}$. So, if

$$A \geq 0, A^2 \geq 0, A^3 \geq 0, \cdots,$$

then

$$\left(I - n\varepsilon|L^{-1}||L|\right)^{-1}_{ij} \geq 0,$$

as long as

$$\boxed{n\varepsilon\left|\lambda_j(|L^{-1}||L|)\right| < 1}$$

We conclude:

$$\begin{aligned}|y-\hat{y}| &\leq \left(I - \underbrace{n\varepsilon|L^{-1}||L|}_{=:A}\right)^{-1}(n+1)\varepsilon|L^{-1}||L||y| \\ &= \left(I + n\varepsilon|L^{-1}||L| + O(n^2\varepsilon^2)\right)(n+1)\varepsilon|L^{-1}||L||y|,\end{aligned}$$

so

$$|y-\hat{y}| \leq (n+1)\varepsilon|L^{-1}||L||y| + O(n^2\varepsilon^2),$$

if $n\varepsilon\lambda_j(|L^{-1}||L|) < 1$.
We can pretty much 'guesstimate' how large this error is by looking at $|L^{-1}|$. So if we get an incorrect, then we say that our model is slightly incorrect (we were handed a bad problem).
Notice that we didn't use the fact that L is lower-triangular, so this is true for all matrices A. Additionally, this applies to if y is not a vector but rather

a matrix. This would tell us about the inverse of a matrix, provided that the Cholesky factors are under control. Recall that in our present case, L has 1s on its diagonal and m_{ij} (multipliers) lower-left entries come from:

$$|m_{ij}| = \left|\frac{a_{ij}}{a_{jj}}\right| \leq 1$$

After the break, we'll talk about matrix norms. Likely tomorrow we'll look at iterative improvement.

Break time.

2 Matrix Norms

Now we want to simplify our previous inequalities which depended on the entire matrices $|L^{-1}|$ and $|L|$. First off, recall the three norms of a vector.

2.1 Review: Vector Norms $||x||_p$

(1) $p = 1$:

$$||x||_p = \sum_{j=1}^{n} |x_j|$$

(2) Euclidean norm ($p = 2$):

$$||x||_p = \sqrt{\sum_{j=1}^{n} x_j^2}$$

(3) $p = \infty$:

$$||x||_p = \max_j |x_j|$$

(Actually, there is a family of norms for different values of p, but we only care about these three for our purposes.)

In the euclidean norm, the 'unit ball' is the standard unit circle. In the max case ($p = \infty$), the 'unit ball' is the unit square. For $p = 1$, the 'unit ball' is the unit diamond.

Hence we say:

$$\boxed{||x||_\infty \leq ||x||_2 \leq \underbrace{||x||_1} \leq \sqrt{n}||x||_2}$$

It turns out we can say more than this, to get the underbraced inequality:

$$\begin{aligned}
||x||_1 &= \sum_{j=1}^{n} |x_j| \cdot 1 \\
&= |x|^T e, \qquad e = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \\
&= ||x||_2 ||e||_2, \qquad \text{Cauchy-Schwarz}, \\
&\leq \sqrt{n}||x||_2,
\end{aligned}$$

where we get the last inequality from $||e||_2 = \sqrt{\underbrace{1+1+\cdots+1}_{n}} = \sqrt{n}$.
We say that all norms are (topologically) **equivalent** in finite-dimensional spaces, in the sense that a sequence converging to 0 in one norm converges to 0 in another norm.

2.2 Matrix Norm

We say that Matrix norms take off from vector norms on a spring-board. Consider A as an operator on a vector $\vec{x}$ (we will simply write x). We may not know how big A is or x is, but we can look at

$$\frac{||Ax||}{||x||}$$

and take the worst case over all x. That is, we can take:

$$||A||_{2,1,\infty} := \max_{x \neq 0} \frac{||Ax||_{2,1,\infty}}{||x||_{2,1,\infty}}$$

Of course, this is not actually how we compute these, because this would require us to check every single x.
Facts:

$$||A_1|| = \max_j \sum_{i=1}^{n} |a_{ij}|,$$

where we take the columns, sum their values, and take the max of these. For example,

$$\left\| \begin{bmatrix} 1 & -2 & 3 \\ -4 & 5 & -6 \\ 7 & -8 & 9 \end{bmatrix} \right\|_1 = \max 12, 15, 18 = 18$$

The dual to this is summing the rows:

$$||A||_\infty := ||A^T||_1 = \max_i \sum_{j=1}^{n} |a_{ij}|,$$

where in our example, the ∞-norm of that matrix is 24.
On the other hand,

$$||A||_2 := \max \text{ eigenvalue of } A^T A = \lambda_{\max}(A^T A)$$

And a weird (but true!) fact is:

$$\boxed{||A||_2^2 \leq ||A||_1 ||A||_\infty},$$

which is worth mentioning because each of the norms on the RHS is easy to compute, whereas computation for $||A||_2^2$ can be difficult or intensive. It's good to know there's an easy-to-compute upper bound for this.
We've found a general class of norms, which we call **operator norms**. The last 1 of 4 norms we consider is **not** an operator norm:

2.3 Frobenius norm

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

We take the absolute value and square in the case that we are over the complex field $\mathbb{C}$, but in our purposes in 128A, we just work in $\mathbb{R}$. Recall the ways we derived to find the Cholesky factorization of a matrix.
(1) Gaussian elimination
(2) Equate all terms and solve a bunch of equations.
(3) Use Newton's method, take an initial guess and update using a matrix uph (upper-triangular, halved)

$$uph(A)_ij := \begin{cases} a_{ij}, & i > j \\ \frac{1}{2}a_{ij}, & i = j \\ 0, & i > j \end{cases}$$

We want the Frobenius norm for the third method above.

Definition: Absolute norm -

We say that a norm is an **absolute norm** if it depends only on the absolute values of the entries.

Surely, the 2-norm is **not** an absolute norm. This is important because we know that when we solve $Ly = c$ in floating point and we get $\hat{y}$ which solves a nearby problem $\hat{L}\hat{y} = \hat{c}$ where $|c - \hat{c}| \leq \varepsilon|c|$ and $|L - \hat{L}| \leq n\varepsilon|L|$, then we proved earlier that:

$$|y - \hat{y}| \leq (n+1)\varepsilon|L^{-1}||L||y| + O(n^2\varepsilon^2).$$

In terms of matrices, if we have an **absolute norm**, then we can simply convert single bars (absolute values) into double bars (norms). Because of the way norms are defined,

$$||Ax|| \leq ||A|||x||,$$

because we took $||A||$ as the maximum of the fraction $\frac{||Ax||}{||x||}$. This is interesting because if we take an eigenvalue with $Ax = \lambda x$, then we have:

$$||Ax|| = ||\lambda x|| = |\lambda|\cancel{||x||} \leq ||A||\cancel{||x||}$$

which tells us:

$$|\lambda| \leq ||A||,$$

where in the 2-norm, we can have equality, but in general we have strict inequality. This means that, for example, if we're looking at an inverse and we want to take the geometric series to expand it, then if $||A|| \leq \frac{1}{2}$, the geometric series works really well. We say this in the same sense that every term of this sequence 'ought to give us an additional bit of accuracy, otherwise we won't want to use it'. Technically, if we have $||A|| < 1$, then this converges (but might be too slow at doing so).

Hence if we take two matrices A, B and multiply them together, we have:

$$\begin{aligned} ||ABx|| &\leq ||A||\ ||Bx|| \\ &\leq ||A||\ ||B||\ ||x||, \end{aligned}$$

and so (for operator norms which are sub-multiplicative):

$$||AB|| \leq ||A||\ ||B||$$

This is why we can do calculations like:

$$|y - \hat{y}| \leq (n+1)\varepsilon |L^{-1}||L||y|,$$

where when we take the norm of a vector, this is simply an absolute norm of the vector. So we write:

$$\frac{||y - \hat{y}||}{||y||} \leq (n+1)\varepsilon \underbrace{||\ |L^{-1}||L|\ ||}_{\text{a number to blame error}},$$

where we can take advantage of the fact that we can divide by the norm $||y||$. Let

$$\boxed{K_{CR}(L) := ||\ |L^{-1}||L|\ ||}$$

be the ***component-wise relative* condition number** of L. If this is is close to 1, within $O(1)$, then we should have no issues and can solve simply. If this is large, then we can look deeper into the problem (or seek to be paid overtime).
Recall from the MVT, we have the following 'condition number'

$$\frac{f(x) - f(y)}{f(x)} = \boxed{\frac{xf'(\xi)}{f(x)}} \cdot \frac{(x-y)}{x},$$

where $\frac{x-y}{x}$ is the relative change in the input, and on the left-hand-side, $\frac{f(x)-f(y)}{f(x)}$ is the relative change in the output. The boxed quantity is the condition number of evaluating f at x, which is an estimate of how 'scary' evaluting near a point is. Immediately, we can see that if we are computing zero and the derivative is nonzero, our condition number is high.
Important: If we derive the above carelessly, we can get get the following '**condition number**' which is a loose bound:

$$\kappa(L) := ||L^{-1}||\ ||L||$$

Notice $\kappa(L) > K_{CR}(L)$.

Lecture ends here.

Tomorrow, we'll look at iterative improvement.