Error Bounds and Condition Estimation.

We have seen that whether LU decomposition yields a backward stable algorithm for solving $Ax = b$ depends on whether the pivot growth factor

$$g_{pp} = \frac{\max |U_{ij}|}{\max |A_{ij}|}$$

is small or grows slowly as a function of $n$.

Proposition. For GEPP, $g_{gepp} \leq 2^{n-1}$.

Proof. When we update $\tilde{A}_{zz}$, we use:

$$\tilde{a}_{jk} = a_{jk} - |j_i| u_{ik}$$

where $|j_i| \leq 1$ but there is no bound on $|u_{ik}|$ except $|u_{ik}| \leq \max |a_{ij}|$. Thus $a_{jk}$ could double on this update. $\square$
It turns out that for complete pivoting, one can show

\[ g_{gecp} \leq \sqrt{n \cdot 2 \cdot 3^{\frac{3}{2}} \cdot 4^{\frac{1}{2}} \cdots \cdot n^{\frac{1}{n-1}}} \approx n^{1/2} + \ln n^{1/4} \]

(see Demmel, p 50).

These give us the error bounds

\[ \|SA\|_\infty \leq 3ne \|L\|_\infty \|U\|_\infty \]
\[ \leq 3ne \cdot n \cdot n \cdot g_{gepp} \|A\|_\infty \]
\[ \leq 3n^3 e 2^{n-1} \|A\|_\infty \]

since the \( L_\infty \) norm of a matrix \( A \) is the largest sum of (abs values of) entries in a row of \( A \), and

\[ \|SA\|_\infty \leq 3n^{3/2} + \ln n^{1/4} \in \|A\|_\infty \]

for GECP by the same argument.
Of course, these error bounds are much too large. A better bound comes from the residual estimate. Recall that if \( \hat{x} \) is an approximate solution to \( Ax = b \), then if

\[
\mathbf{r} = A\hat{x} - b
\]

we can estimate

\[
\Delta x = \hat{x} - x = A^{-4}(r+b) - A^{-4}b
\]

but

\[
\|\Delta x\| \leq \|A^{-4}\| \|r\|. 
\]

Now it is easy and cheap to compute \( \mathbf{r} \) and \( \|\mathbf{r}\| \), but how can we estimate \( \|A^{-4}\| \)? It is too slow to compute \( A^{-4} \) directly and then calculate its norm.
Instead, we try to use the LU decomposition of $A$ to estimate $\|A^{-1}\|$.

Idea. We want to estimate $\|B\|_1$ for a matrix $B$. By definition,

$$\|v\|_1 = \sum |v_i|$$

and

$$\|B\|_1 = \max_{x \neq 0} \frac{\|Bx\|_1}{\|x\|_1} = \max \sum_{i=1}^{n} |b_{ij}|$$

So one strategy would be to compute columns of $B = A^{-1}$ and measure their one-norms.

To compute a column, we must compute

$$B e_j = A^{-1} e_j = \hat{x}$$

or equivalently, to solve

$$A \hat{x} = e_j$$

which is an $O(n^2)$ operation given the LU decomposition of $A$. 
Thus computing all $n$ columns would be again $O(n^3)$.

Observations

$$
\|B\|_1 = \max_{\|x\|_1 \leq 1} \|Bx\|_1.
$$

$$
\{ x \mid \|x\|_1 \leq 1 \} \text{ is convex.}
$$

$$
f(x) = \|Bx\|_1 \text{ is a convex function.}
$$

Check: $f(\alpha x + (1-\alpha) y) = \|\alpha Bx + (1-\alpha) By\|_1$

$$
\leq \alpha \|Bx\|_1 + (1-\alpha) \|By\|_1
\leq \alpha f(x) + (1-\alpha) f(y).
$$

(This is just the triangle inequality.)

Our plan is to apply a numerical method to maximize $f(x)$ on $\{ x \mid \|x\|_1 \leq 1 \}$.  

\[\text{(5)}\]
Now for any convex function

\[ f(y) - f(x) \geq \nabla f(x) \cdot (y-x) \]

we have

\[ f(x) + \nabla f(x) \cdot (y-x) \leq f(y) \]

So we can get a lower bound on \( f(y) \) by

\[ f(x) + \nabla f(x) \cdot (y-x) \leq f(y) \]

We will step according to this method (gradient ascent). But how do we compute \( \nabla f \)?

\[ f(x) = \sum_i |\sum_j b_{ij} x_j| \]

if \( \sum_j b_{ij} x_j \neq 0 \), let \( s_i = \text{sign} \sum_j b_{ij} x_j = \pm 1 \).
So

\[ f(x) = \sum \sum s_{ij} b_{ij} x_j \]

and

\[ \frac{\partial f}{\partial x_k} = \sum \sum s_i b_{ik} \]

so if \( S \) is the vector of signs

\[ \nabla f = S^T B = (B^T S)^T \]

Now if \( A = LU \) then \( A^T = U^T L^T \) so \( U^T L^T \) is the LU decomposition of \( A^T \) (the upper triangular matrix \( L^T \) is now the unit one, but who cares).

So we can state find

\[ (A^{-1})^T S = X \Rightarrow (A^T)^{-1} S = X \]

\[ \Rightarrow S = A^T X \]

by solving \( S = A^T X \) using \( A^T = U^T L^T \).
We are left with

Algorithm (Hager's condition estimator)

Choose any \( x \) with \( \|x\|_1 = 1 \):

repeat 3

let \( w = Bx, \ S = \text{sign}(w), \ z = B^T S \)

if \( \|z\|_{\infty} \leq z^T x \) then

return \( \|w\|_1 \)

else

set \( x = e_j \) where \( |z_j| = \|z\|_{\infty} \)

3

This is a little opaque when you first see it, so let's prove it works.
Theorem. When \( \|w\|_1 \) is returned, \( \|w\|_1 = \|Bx\|_1 \) is a local max of \( f(x) = \|Bx\|_1 \). Otherwise \( \|B_{e_1}\| > \|Bx\| \) so the algorithm has made progress.

Proof. Suppose \( \|w\|_1 \) is returned. We know \( \|z\|_\infty \leq z^T x \). Now near \( x \) (as long as we don't change any signs),

\[
f(x) = \|Bx\|_1 = \sum_{i,j} S_{ij} b_{ij} x_j
\]

is linear in \( x \) so

\[
f(y) = f(x) + \nabla f(x) \cdot (y-x)
\]

Now suppose \( y \) is near \( x \) and \( \|y\|_1 = 1 \). We want

\[
\nabla f(x) \cdot (y-x) = z^T (y-x) \leq 0.
\]
But
\[ z^T (y-x) = z^T y - z^T x \]
\[ = \Sigma z_i y_i - z^T x \]
\[ \leq \Sigma |z_i| |y_i| - z^T x \]
\[ \leq \|z\|_\infty \|y\|_1 - z^T x \]
\[ \leq \|z\|_\infty - z^T x \leq 0 \quad \text{as desired.} \]

Now suppose \( \|z\|_\infty > z^T x \). We must show that if \( \hat{x} = e_j \cdot \text{sign}(z_j) \) where \( |z_j| = \|z\|_\infty \), then
\[
\begin{align*}
f(\hat{x}) &\geq f(x) + \nabla f(x) \cdot (\hat{x} - x) \\
&\geq f(x) + z^T (\hat{x} - x) \\
&\geq f(x) + z^T \hat{x} - z^T x \\
&\geq f(x) + |z_j| - z^T x \\
&\geq f(x) + \|z\|_\infty - z^T x \\
&> f(x).
\end{align*}
\]

Experiments show that this is generally within a factor of 2 of the true condition number.

How do we use this in practice? We know

\[
\text{error (in each entry)} = \frac{\|\delta x\|_\infty}{\|x\|_\infty} \leq \frac{\|A^{-1}\|_\infty}{\|x\|_\infty} \frac{\|r\|_\infty}{\|\hat{x}\|_\infty}
\]

Now \(\|A^{-1}\|_\infty = \max_{\text{row}} \text{sum while}\)
\(\|A\|_1 = \max_{\text{column}} \text{sum so if we apply the condition estimator to find}\)
\(\| (A^{-1})^T \|_1 = \| A^{-1}\|_\infty, \) we can compute the rhs estimate.