NA Slides
Aaron Naiman

(based on
Numerical Mathematics and
Computing—Cheney/Kincaid)

Linears Systems

- $\Rightarrow$ Introduction
- Naive Gaussian Elimination
- Limitations
- Operation Counts
- Additional Notes

What Are Linear Systems (LS)?

$$\begin{array}{ccccccccc} a_{1\,1}\,x_1 & + & a_{1\,2}\,x_2 & + & \cdots & + & a_{1\,n}\,x_n & = & b_1 \\ a_{2\,1}\,x_1 & + & a_{2\,2}\,x_2 & + & \cdots & + & a_{2\,n}\,x_n & = & b_2 \\ \vdots & + & \vdots & + & \ddots & + & \vdots & = & \vdots \\ a_{m\,1}\,x_1 & + & a_{m\,2}\,x_2 & + & \cdots & + & a_{m\,n}\,x_n & = & b_m \end{array}$$

- Dependence on unknowns: powers of degree ≤ 1
- Summation form: $\displaystyle\sum_{j=1}^{n} a_{i\,j}\,x_j = b_i$, $1 \leq i \leq m$, i.e., m equations
- Presently: $m = n$, i.e., square systems (later: $m \neq n$)

$\boxed{\textit{Q}\text{: How to solve for } [x_1 \;\; x_2 \;\; \ldots \;\; x_n]^T\text{? } \textit{A}\text{: } \ldots}$

Linears Systems

- Introduction

$\Rightarrow$ Naive Gaussian Elimination

- Limitations
- Operation Counts
- Additional Notes

Overall Algorithm and Definitions

- Currently: direct methods only (later: iterative methods)

- General idea:
  * Generate upper triangular system
    (“forward elimination”)
  * Easily calculate unknowns in reverse order
    (“backward substitution”)

- “Pivot row” = current one being processed
  “pivot” = diagonal element of pivot row

$\boxed{\text{Steps applied to RHS as well.}}$

Forward Elimination

- Generate zero columns below diagonal

  for each row $i := 1, n$ { // process rows downward

      for each row $k := i + 1, n$ { // $\forall$ rows below pivot

          multiply pivot row $\ni$ $a_{i\,i} = a_{k\,i}$

          subtract pivot row from row$_k$

      } // now column below $a_{i\,i}$ is zero

  } // now $a_{i\,j} = 0$, $\forall i > j$

- Obtain triangular system

$\boxed{\text{Let's work an example, ...}}$

Compact Form of LS

$$\left.\begin{array}{rrrrrrrrrr} 6\,x_1 & - & 2\,x_2 & + & 2\,x_3 & + & 4\,x_4 & = & 16 \\ 12\,x_1 & - & 8\,x_2 & + & 6\,x_3 & + & 10\,x_4 & = & 26 \\ 3\,x_1 & - & 13\,x_2 & + & 9\,x_3 & + & 3\,x_4 & = & -19 \\ -\;6\,x_1 & + & 4\,x_2 & + & 1\,x_3 & - & 18\,x_4 & = & -34 \end{array}\right\} \rightarrow$$

$$\left(\begin{array}{rrrr|r} 6 & -2 & 2 & 4 & 16 \\ 12 & -8 & 6 & 10 & 26 \\ 3 & -13 & 9 & 3 & -19 \\ -6 & 4 & 1 & -18 & -34 \end{array}\right)$$

$\boxed{\text{Proceeding with the forward elimination, \ldots}}$

Forward Substitution—Example

$$\left(\begin{array}{rrrr|r} 6 & -2 & 2 & 4 & 16 \\ 12 & -8 & 6 & 10 & 26 \\ 3 & -13 & 9 & 3 & -19 \\ -6 & 4 & 1 & -18 & -34 \end{array}\right) \rightarrow \left(\begin{array}{rrrr|r} 6 & -2 & 2 & 4 & 16 \\ 0 & -4 & 2 & 2 & -6 \\ 0 & -12 & 8 & 1 & -27 \\ 0 & 2 & 3 & -14 & -18 \end{array}\right) \rightarrow$$

$$\left(\begin{array}{rrrr|r} 6 & -2 & 2 & 4 & 16 \\ 0 & -4 & 2 & 2 & -6 \\ 0 & 0 & 2 & -5 & -9 \\ 0 & 0 & 4 & -13 & -21 \end{array}\right) \rightarrow \left(\begin{array}{rrrr|r} 6 & -2 & 2 & 4 & 16 \\ 0 & -4 & 2 & 2 & -6 \\ 0 & 0 & 2 & -5 & -9 \\ 0 & 0 & 0 & -3 & -3 \end{array}\right)$$

Matrix is upper triangular.

Backward Substitution

$$\left(\begin{array}{rrrr|r} 6 & -2 & 2 & 4 & 16 \\ 0 & -4 & 2 & 2 & -6 \\ 0 & 0 & 2 & -5 & -9 \\ 0 & 0 & 0 & -3 & -3 \end{array}\right)$$

- Last equation: $-3x_4 = -3 \Rightarrow x_4 = 1$
- Second to last equation: $2x_3 - 5\underbrace{x_4}_{=1} = 2x_3 - 5 = -9 \Rightarrow$ $x_3 = -2$
- ... second equation ... $x_2 = \ldots$
- ... $[x_1\ x_2\ x_3\ x_4]^T = [3\ 1\ -2\ 1]^T$

For small problems, check solution in original system.

Linears Systems

- Introduction
- Naive Gaussian Elimination

$\Rightarrow$ Limitations

- Operation Counts
- Additional Notes

Zero Pivots

- Clearly, zero pivots prevent forward elimination
- ⚠ zero pivots can appear along the way
- Later: When guarenteed no zero pivots?
- All pivots $\neq 0 \overset{?}{\Rightarrow}$ we are safe

$\boxed{\text{Experiment with system with known solution.}}$

Vandermonde Matrix

$$\begin{pmatrix} 1 & 2 & 4 & 8 & \cdots & 2^{n-1} \\ 1 & 3 & 9 & 27 & \cdots & 3^{n-1} \\ 1 & 4 & 16 & 64 & \cdots & 4^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & n+1 & (n+1)^2 & (n+1)^3 & \cdots & (n+1)^{n-1} \end{pmatrix}$$

- Want row sums on RHS $\Rightarrow x_i = 1,\ i = 1, \ldots, n$
- Define: $p(t) \equiv 1 + t + t^2 + \cdots + t^{n-1} = \sum_{j=1}^{n} t^{j-1}$, $t := 1 + i$
- By geometric series we obtain b_i

$$\sum_{j=1}^{n} \underbrace{(1+i)^{j-1}}_{a_{i\,j}} \cdot \underbrace{1}_{x_j} = \frac{(1+i)^n - 1}{(1+i) - 1} = \underbrace{\frac{1}{i}[(1+i)^n - 1]}_{b_i}$$

$\boxed{\text{System is ready to be tested.}}$

Vandermonde Test

- Platform with 7 significant (decimal) digits
  - $n = 1, \ldots, 8 \Rightarrow$ expected results
  - $n = 9$: error $>$ 16,000% !!

- Questions:
  - What happened?
  - Why so sudden?
  - Can anything be done?

- Answer: matrix is "ill-conditioned"
  - Sensitivity to roundoff errors
  - Leads to error propagation and magnification

$\boxed{\text{First, how to assess vector errors.}}$

Errors

- Given system: $Ax = b$ and solution estimate $\tilde{x}$
- Residual (error): $r \equiv A\tilde{x} - b$
- Absolute error (if x is known): $e \equiv x - \tilde{x}$
- Norm taken of r or e: vector $\rightarrow$ scalar quantity (more on norms later)
- Relative errors: $||r||/||b||$ and $||e||/||x||$

Back to ill-conditioning, ...

Ill-conditioning

- $\left.\begin{array}{rcrcl} 0 \cdot x_1 & + & x_2 & = & 1 \\ x_1 & + & x_2 & = & 2 \end{array}\right\} \Rightarrow$ 0 pivot

- General rule: if 0 is problematic $\Rightarrow$
  numbers near 0 are problematic

- $\left.\begin{array}{rcrcl} \epsilon\, x_1 & + & x_2 & = & 1 \\ x_1 & + & x_2 & = & 2 \end{array}\right\} \ldots\ x_2 = \frac{2-1/\epsilon}{1-1/\epsilon}$ and $x_1 = \frac{1-x_2}{\epsilon}$

- ϵ small (e.g., $\epsilon = 10^{-9}$ with 8 significant digits) $\Rightarrow x_2 = 1$ and $x_1 = 0$—wrong!

$\boxed{\text{What can be done?}}$

Pivoting

- Switch order of equations, moving offending element off diagonal
- $\left.\begin{array}{rcrcl} x_1 & + & x_2 & = & 2 \\ \epsilon\, x_1 & + & x_2 & = & 1 \end{array}\right\} \Rightarrow$, $x_2 = \frac{1-2\epsilon}{1-\epsilon}$ and $x_1 = 2 - x_2 = \frac{1}{1-\epsilon}$
- This is correct, even for small ϵ
- Compare size of diagonal (pivot) elements to ϵ
- Ratio of first row of Vandermonde matrix $= 1 : 2^{n-1}$

Issue is relative size, not absolute size.

Linears Systems

- Introduction
- Naive Gaussian Elimination
- Limitations

$\Rightarrow$ Operation Counts

- Additional Notes

How Much Work on A?

- Counting $\times$ and $\div$ (i.e., long operations) only

- Pivoting: row decision amongst k rows = k ratios

- First row:
  - n ratios
  - $n-1$ multipliers
  - $(n-1)^2$ multiplications

  total: n^2 operations

- $\therefore$ forward elimination operations (for large n)

$$\sum_{k=2}^{n} = \frac{n}{6}(n+1)(2n+1)+1 \approx \frac{n^3}{3}$$

$$\boxed{\text{How about the work on } b\text{?}}$$

Operation Count for RHS

- Forward elimination: $\sum_{k=2}^{n} (k-1) = \dfrac{n(n-1)}{2}$
- Backward substitution: $\sum_{k=1}^{n} k = \dfrac{n(n+1)}{2}$
- Total: n^2 operations
- $O(n)$ fewer operations than on A
- Important for multiple RHSs

$\boxed{\text{Can we do better at times?}}$

Sparse Systems

$$
\begin{pmatrix}
\times & \times & 0 & \cdots & & \cdots & 0 \\
\times & \times & \ddots & \ddots & & & \vdots \\
0 & \ddots & \ddots & & & & \\
\vdots & \ddots & & & & \ddots & \vdots \\
 & & & & & \ddots & 0 \\
\vdots & & & \ddots & \ddots & \ddots & \times \\
0 & \cdots & & \cdots & 0 & \times & \times
\end{pmatrix}
$$

- Above, e.g., tridiagonal system (half bandwidth $= 1$)
- Opportunities for savings
    - $*$ storage
    - $*$ computations
- Both are $O(n)$

Linears Systems

- Introduction
- Naive Gaussian Elimination
- Limitations
- Operation Counts

$\Rightarrow$ Additional Notes

Pivot-Free Guarantee

- When are we guaranteed non-zero pivots?
- Diagonal dominance (just like it sounds):

$$|a_{i\,i}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} \left|a_{i\,j}\right|, \quad i = 1, \ldots, n$$

- (Or "$>$" in one row, and "$\geq$" in remaining)
- Many finite difference and finite element problems $\Rightarrow$
  diagonally dominant systems

$\boxed{\text{Occurs often enough to justify individual study}}$

LU Decomposition

- E.g.: same A, many b's of time-dependent problem
- Want $A = LU$ for decreased work later
- Then define y: $L\underbrace{Ux}_{\equiv y} = b$
  - $*$ solve $Ly = b$ for y
  - $*$ solve $Ux = y$ for x
- U is upper triangular, result of Gaussian elimination
- L is unit lower triangular, 1's on diagonal and Gaussian multipliers below

$$\boxed{\text{Each new RHS is } n^2 \text{ work, instead of } O\left(n^3\right)}$$