IE170: Algorithms in Systems Engineering: Lecture 29

Jeff Linderoth

Department of Industrial and Systems Engineering
Lehigh University

April 16, 2007

Taking Stock

Last Time

- Matrix Review

This Time

- Solving Triangular Systems
- Solving Symmetric Positive Definite Systems
- Least Squares

Systems of Equations: $Ax = b$

- From our previous discussion, we know that the system of equations $Ax = b$ has a unique solution if and only if the matrix A is square and invertible
- This is true if the columns A_j are linearly independent
- From now on, we will consider only invertible systems.
- In fact, today we will consider special versions of A

The $64 Question

- How do we solve a systems of equations?
- We factor the matrix A into a simpler form

Triangular Systems

- Let's suppose that we are able to find two $n \times n$ matrices L, U such that
$$A = LU$$
where
  - L is upper triangular.
  - U is lower triangular with 1's on the diagonal.
- How could use such a decomposition to solve the system $Ax = b$?

Using a Triangular Decomposition

- Once we have an triangular decomposition, we can use it to easily solve the system $Ax = b$.
- Note that the system $Ax = b$ is equivalent to the original system, which is then equivalent to $LUx = b$.
- We can solve the system in two steps:
  - First solve the system $Ly = b$ (forward substitution).
  - Then solve the system $Ux = y$ (backward substitution).

Forward Substitution

$$\begin{array}{lllllllll} \ell_{11}y_1 & & & & & & & = & b_1 \\ \ell_{21}y_1 & + & \ell_{22}y_2 & + & \cdots & & & = & b_2 \\ \ell_{n1}y_1 & + & \ell_{n2}y_2 & + & \ell_{n3}y_3 & + & \cdots & = & b_n \end{array}$$

- Just substitute forward into:

$$y_i = \frac{b_i - \sum_{j=1}^{i-1} \ell_{ij} y_j}{\ell_i i}$$

- So we have y such that $Ly = b$.

Example Matrix

- Next, we simply solve the system $Ux = y$
- Backwards substitution works in a similar fashion, but loops "down" from n down to 1

$$\begin{array}{lllllllll} u_{11}x_1 & + & u_{12}x_2 & + & u_{13}x_3 & + & \cdots & = & y_1 \\ & & u_{22}x_2 & + & u_{23}x_3 & + & \cdots & = & y_2 \\ & & & & & & u_{nn}x_n & = & y_n \end{array}$$

TriangularSolve(L, U, b)

```
n ← rows[L]
for i ← 1 to n
do y[i] ← (b[i] − Σ_{j<i} ℓ_ij y_j)/ℓ_ii
for i ← n to 1
do x[i] ← (y[i] − Σ_{j>i} u_ij x_j)/u_ii
return x
```


Example

$$L = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 1 & 3 & 0 & 0 \\ 2 & 2 & 2 & 0 \\ 1 & 1 & 3 & 1 \end{bmatrix} \quad U = L^T \quad b = \begin{bmatrix} 32 \\ 26 \\ 28 \\ 30 \end{bmatrix}$$

- We'll solve $Ax = LUx = b$.
- Hopefully we get $x = (1, 1, 1, 1)^T$

Special Matrices

- If A is a square symmetric ($A = A^T$) matrix such that

$$x^T A x > 0 \ \forall x \in \mathbb{R}^n, x \neq 0$$

then A is said to be symmetric positive definite.

They are everywhere!

- Electrical circuit problems
- Structural Engineering (elastic deformations)
- Variance-Covariance matrices
- Numerical solution of partial differential equations
- Solution of linear systems
  - $Ax = b \Leftrightarrow x$ minimizes $1/2x^T Ax - b^t x$
- Least Squares Problems!

Fun SPD Facts

- If A is SPD, then A is nonsingular
  - Proof: If A is singular, then there is $x \neq 0 \in \mathbb{R}^n$ with $Ax = 0$, so $x^T Ax = 0$, and x is not SPD. QUITE ENOUGH DONE
- If M is non-singular, then $A = MM^T$ is SPD
  - Proof: $A^T = (MM^T)^T = MM^T = A$, so A is symmetric. $x^T Ax = x^T(MM^T)x = y^T y$ for $y = M^T x$. $y^T y = \sum_{i=1}^n y_i^2 \geq 0$, and it is only 0 when $y = 0$, but $y \neq 0$ or else M^T would have been singular, since $y = M^T x$. QUITE ENOUGH DONE

Looking for a Decomposition

- Wouldn't it be awesome if $U = L^T$ (like our example), so there was a decomposition of the form $A = LL^T$. Let's check to see if that is possible:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} \ell_{11} & 0 & \cdots & 0 \\ \ell_{21} & \ell_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \ell_{n1} & \ell_{n2} & \cdots & \ell_{nn} \end{bmatrix} \begin{bmatrix} \ell_{11} & \ell_{21} & \cdots & \ell_{n1} \\ 0 & \ell_{22} & \cdots & \ell_{n2} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \ell_{nn} \end{bmatrix}$$

- And this implies that...
  - $a_{i1} = \ell_{i1}\ell_{11}$, so $a_{11} = \ell_{11}^2$, $\ell_{i1} = a_{i1}/\ell_{11}$
  - $a_{i2} = \ell_{i1}\ell_{21} + \ell_{i2}\ell_{22}$

General Formula

In General

$$a_{ij} = g_{i1}g_{j1} + g_{i2}g_{j2} + \ldots + g_{i,j-1}g_{j,j-1} + g_{ij}g_{jj}$$

- This only depends on columns up to j!
- Assuming we have computed the first $j-1$ columns of L, the j^{th} columns can be computed using the formulae

$$g_{jj} = \sqrt{a_{jj} - \sum_{k<j} g_{jk}^2}$$

$$g_{ij} = \frac{a_{ij} - \sum_{k<j} g_{ik}g_{jk}}{g_{jj}} \quad \text{for } j > i$$

Query and Example

- What if $a_{jj} - \sum_{k<j} g_{jk}^2 < 0$?
- Then A is not SPD.
- The proof of this fact is too complicated to give now, but it is true that A is SPD if and only if it can be written as $A = LL^T$ for a lower triangular matrix L
- L is known as the Cholesky factor of A, after the French mathematician André-Louis Cholesky.

$$A = \begin{bmatrix} 16 & 4 & 8 & 4 \\ 4 & 10 & 8 & 4 \\ 8 & 8 & 12 & 10 \\ 4 & 4 & 10 & 12 \end{bmatrix}$$

- Assuming we can do the arithmetic correctly, we should get $A = LL^T$, with L the previous L in this lecture.

Least Squares

- Suppose I am given some data points (measurements)

$$(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m)$$

And we wish to find a function that closely approximates these measurements:

$$y_i = F(x_i) + \eta_i$$

where the η_i are "small."

- We will assume that $F(x)$ has the form:

$$F(x) = \sum_{j=1}^{n} c_j f_j(x)$$

Least Squares

- A common choice of the "basis functions" $f_j(x)$ are small order polynomials:

$$F(x) = c_1 + c_2 x + c_3 x^2 + \ldots + c_n x^{n-1}.$$

- Choosing $n = m$ means that the function will exactly match the y_i, and is generally a "bad idea", as this is known as overfitting,
- Instead, n is typically much smaller than m
  - For example, if $n = 2$, then we are looking for the best "linear" fit of the data

More Least Squares

- Let's create the matrix

$$A = \begin{bmatrix} f_1(x_1) & f_2(x_1) & \vdots & f_n(x_1) \\ f_1(x_2) & f_2(x_2) & \vdots & f_n(x_2) \\ \vdots & \vdots & \ldots & \vdots \\ f_n(x_1) & f_n(x_1) & \vdots & f_n(x_1) \end{bmatrix}$$

- So $Ac = [F(x_1), F(x_2), \ldots F(x_m)]^T$ is the m-vector of predicted values for y, so
- $\eta = Ac - y$ is the vector that we are trying to minimize

Least Squares

- In least-squares, we minimize the squared (Euclidean) length of η, or

$$\min \|\eta\|^2 = \|Ac - y\|^2 = \sum_{i=1}^{m} \left(\sum_{j=1}^{n} a_{ij} c_j - y_i \right)^2$$

- Taking derivatives, setting the result equal to zero, and putting things back in matrix notation, means that we look for a c such that

$$(Ac - y)^T A = 0 \text{ or } A^T Ac = A^T y.$$

Solving Least Squares

Normal Equations

$$A^T Ac = A^T y$$

- We seek $c = (A^T A)^{-1} A^T y$
- Sometimes $A^+ \stackrel{\text{def}}{=} (A^T A)^{-1} A^T$ is call the pseudoinverse of A, and it exists for non-square A
- We don't really nned to take the inverse, we just solve the SPD system $A^T Ac = A^T y$
- This is what you get to do in lab today!