QR Matrix Factorization
Numerical Stability

CS 111: Introduction to Computational Science

Spring 2019 Lecture #5

Ziad Matni, Ph.D.

Administrative

- Homework #2 due
- Homework #3 due next Monday

Lecture Outline

- Review: Cholesky Factorization
- QR Factorization
- Numerical Stability

SPD Matrices

<u>SPD = Symmetrical Positive Definite</u>

- SPD Matrices are the equivalent of "positive numbers"
  - When you multiply SPD Matrices with a vector, its direction stays "similar"

- SPDs show up a lot in statistics, control system designs, heat conductivity designs, etc…

- There are many good algorithms (**fast**, **numerically stable**) that work better for an SPD matrix, such as Cholesky factorization.

Characteristics of SPD Matrices

- $x^{\mathrm{T}}\mathbf{A}x > 0$ for every non-zero x
- Mathematically also means that **all** its eigenvalues are > 0
- LU factorization without pivoting succeeds, with all pivots > 0

Cholesky Factorization

- For specific cases where we have:
  - SPD **square** matrix, **A**
  - So, all the ***eigenvalues*** of **A** are positive and $x^T\mathbf{A}x > 0$ for every non-zero vector x
- We can factor **A** into $\mathbf{X}^T\mathbf{X}$
- Cholesky factorization is a particular form where **X** is an upper triangular with positive diagonals (called **R**)
- So, if you can find **R**, you can figure out: $\mathbf{A} = \mathbf{R}^T\mathbf{R}$
- See last week's lecture for an example

Recall: What are the Eigenvalues of…

For example,

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

Is A a SPD Matrix?
Is $x^T.A.x > 0$?

$\lambda 1 + \lambda 2 = a + d = 2 + 2 = 4$ (aka "trace" of A)
$\lambda 1 . \lambda 2 = ad - bc = 4 - 1 = 3$

So, $\lambda 1 = 4 - \lambda 2$ and $(4 - \lambda 2).\lambda 2 = 3$
And after some math… $\lambda 1 = 1$ & $\lambda 2 = 3$

QR Factorization

- Often used to solve the **linear least squares problem**
  - An approximation of linear functions to data.
  - Re: solving statistical problems in **linear regression**

QR Factorization

- **A = QR**, where **Q** is an orthogonal matrix based on **A**
  - Orthogonal matrix → It's columns are "orthonormal"
  - Property: $\mathbf{Q^TQ = QQ^T = I}$
- **R** is an upper triangular matrix
  - Property: If $A = QR$ → $Q^TA = Q^TQR$ → $\mathbf{R = Q^TA}$
- See blackboard exercise using the *Gram-Schmidt process*

Numerical Stability

- A generally desirable property of numerical algorithms

- Consider **f(x) = y** (mathematically) that is calculated to be **y*** algorithmically/computationally
  - **y*** is a ***deviation*** from the "true" solution **y**
    b/c of round-off errors and/or truncation errors

- **Forward Error**: $\Delta y = y^* - y$
- **Backward Error**: Smallest Δx such that $f(x + \Delta x) = y^*$
- **Relative Error**: $|\Delta x| / |x|$

Numerical Stability

- In a computational matrix, we look at a matrix **A** and how small changes in it can lead to either small or large changes in calculations
  - You want small changes to yield small changes – that's "stability"…

Example:

$$A = \begin{bmatrix} 1 & 1000 \\ 0 & 1 \end{bmatrix} \quad \text{and } Ax = b \text{ where } b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

- See blackboard for details and for counter-examples

Numerical Stability

- Forward and backward error are related by the **condition number**
  - We'll examine its calculation another time
  - You can use **numpy** to calculate it!

- Large condition number = the matrix is not numerically stable
  - Or, "ill-conditioned"
- Small condition number (closer to 1) = the matrix **is** numerically stable
  - Or, "well-conditioned"

- Usually, symmetrical and/or is normal matrices are "well-conditioned"

Your To-Dos

- Homework #3 – due **Monday, April 22nd**

</LECTURE>