Announcements

1) Ignore "When Vectors Become Continuous Functions" Section in Lecture 7.

2) 2 new files on classical + "modified" Gram-Schmidt for QR decomposition on Canvas

Algorithm for QR Decomposition

Matlab!

A a matrix, in Matlab,

A(:, j) returns the jth column of A.

$A(i,:)$ returns the i^{th} row.

QR factorization through Classical Gram-Schmidt Algorith

See Matlab program "clgs.m". Define A, type $[Q,R]=clgs(A)$.

QR Factorization through "modified" Gram-Schmidt Algorithm

This is what we did in the 2×2 matrix example yesterday.

"Triangular Orthogonalization"

vs.

"Orthogonal Triangularization"

Gram-Schmidt Projections

$$A = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix}$$

$$P_1 = I_m, \qquad q_1 = a_1 / \|a_1\|_2$$

$$P_2 = I_m - P_{\text{span}(\{q_1\})}, \qquad q_2 = P_2 a_2 / \|P_2 a_2\|_2$$

$$P_3 = I_m - P_{\text{span}\{q_1, q_2\}}, \qquad q_3 = P_3 a_3 / \|P_3 a_3\|_2$$

In general, for $2 \le j \le n$

$$P_j = I_m - P_{\text{span}\{a_1, a_2, \cdots, a_{j-1}\}}$$

$$q_j = P_j a_j / \| P_j a_j \|_2$$

(assuming $\text{rank}(A) = n$)

These are the columns of Q in the QR decomposition of A.

Classical vs. Modified

Classical computes P_j,

then uses it to compute q_j.

Modified computes

$$P_{\{a_{j-1}\}^\perp} \cdots P_{\{a_2\}^\perp} P_{\{a_1\}^\perp}$$

$$(= P_j)$$

There's no difference in the computation – its the way in which the computation takes place

Computational Cost

Definition: (flop) A flop is a floating point operation.

This refers to any of the traditional operations on numbers: addition, subtraction, multiplication, or division.

Any such instance of an operation involving two numbers counts as one flop. Additionally, taking a square root counts as one flop.

Example 1: (2 norm, determinant)

$$v = \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

Calculate $\|v\|_2$.

$$\|v\|_2 = \overbrace{\sqrt{\underbrace{1\cdot 1}_{\text{flop}} \underset{\text{flop}}{+} \underbrace{(-1)\cdot(-1)}_{\text{flop}}}}^{\text{flop}}$$

four flops on a computer.

In general, if $v \in \mathbb{C}^n$,

calculating $\|v\|_2$

takes $\underbrace{n}_{\text{multiplying}} + (\underbrace{n-1}_{\text{adding}}) + \underbrace{1}_{\text{square root}}$

$= 2n$ flops.

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

$$\det(A) = 1 \cdot 4 - 2 \cdot 3$$

$$= 3 \text{ flops}$$

As the size of the matrix grows, the number of flops gets much bigger!