Sergio Garcia Tapia
Numerical Linear Algebra, Lloyd Trefethen and David Bau III
Lecture 8: Gram Schmidt Factorization
June 23, 2024

Lecture 8: Gram Schmidt Factorization

Exercise 1. Let A be an $m \times n$ matrix. Determine the exact numbers of floating point additions, subtractions, and multiplications involved in computing the factorization $A = \hat{Q}\hat{R}$ by Algorithm 8.1.

Solution: The presented algorithm was:

```
# Initialize v_i to columns of A
for i = 1 to n
    v_i = a_i

# Create orthogonal list from columns of A
for i = 1 to n
    r_ii = ||v_i|| # Norm of v_i
    q_i = v_i/r_ii # Normalize
    for j = i + 1 to n
        r_ij = adj(q_i) * v_j # inner product of q_i and v_j
        v_j = v_j - r_ijq_i
```

The norm operation,$\|v_i\|$, involves m products (squaring each entry), $m-1$ addition (to add the m products), and one square root. The expression $q_i = v_i/r_{ii}$ involves m divisions. Each of these two operations occurs exactly n times, since they occur in the outer loop.

As discussed in page 59, the operations of the inner loop are

$$\begin{aligned} r_{ij} &= q_i^* v_j \\ v_j &= v_j - r_{ij} q_i \end{aligned}$$

Since $a_i \in \mathbb{C}^m$, the inner product $q_i^* v_j$ involves m multiplications (one for each pair of entries) and $m-1$ additions (one for each pair of the m multiplication results). Meanwhile, $r_{ij}q_i$ involves m multiplications, and subtracting the result from v_j in the expression $v_j - v_j - r_{ij}q_i$ involves m subtractions. Thus the exact number of operations in each iteration of the inner loop is $4m-1$: $2m$ multiplications, m subtractions, and $m-1$ additions.

Note that each inner loop operations occurs a constant number of times, so its helpful to compute the sum:

$$\begin{aligned}
\sum_{i=1}^{n} \sum_{j=i+1}^{n} (1) &= \sum_{i=1}^{n} (n-i) \\
&= \sum_{i=1}^{n} (n) - \sum_{i=1}^{n} i \\
&= n^2 - \frac{n(n+1)}{2} \\
&= \frac{2n^2}{2} - \frac{n^2+n}{2} \\
&= \frac{n(n-1)}{2}
\end{aligned}$$

Altogether:

$$\begin{aligned}
\text{Square Roots} &: 1 \\
\text{Divisions} &: n \\
\text{Addition} &: n \cdot (m-1) + \sum_{i=1}^{n} \sum_{j=i+1}^{n} (m-1) \\
&= n(m-1) + \frac{(m-1)n(n-1)}{2}. \\
&= \frac{(m-1)n(n+1)}{2} \\
\text{Subtractions} &: \sum_{j=i+1}^{n} (m) = \frac{mn(n-1)}{2} \\
\text{Multiplications} &: m + \sum_{j=i+1}^{n} (2m) = m + mn(n-1) \\
&= m(n^2 - n + 1)
\end{aligned}$$

Exercise 8.2. Write a MATLAB function `[Q,R] = mgs(A)` (see next lecture) that computes a reduced QR factorization $A = \hat{Q}\hat{R}$ of an $m \times n$ matrix with $m \geq n$ using modified Gram Schmidt orthogonalization. The output variables are a matrix $Q \in \mathbb{C}^{m \times n}$ with orthonormal columns and a triangular matrix $R \in \mathbb{C}^{m \times n}$.

Exercise 3. Each upper-triangular matrix R_j of p. 61 can be interpreted as the product of a diagonal matrix and a unit upper-triangular matrix (i.e., an upper-triangular matrix with 1 on the diagonal). Explain exactly what these factors are, and which line of Algorithm 8.1 corresponds to each.

Solution: Consider the example matrix given in page 61 for $R_1 \in \mathbb{C}^{n \times n}$:

$$R_1 = \begin{bmatrix} \frac{1}{r_{11}} & \frac{-r_{12}}{r_{11}} & \frac{-r_{13}}{r_{11}} & \cdots \\ & 1 & & \\ & & 1 & \\ & & & \ddots \end{bmatrix}$$

Then $R_1 = D_1 T_1$, where $D_1, T_1 \in \mathbb{C}^{n\times n}$, D_1 is a diagonal matrix, and T_1 is an upper triangular matrix of the form

$$D_1 = \begin{bmatrix} \frac{1}{r_{11}} & 0 & \cdots & 0 \\ 0 & 1 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & & 1 \end{bmatrix} \qquad T_1 = \begin{bmatrix} 1 & -r_{12} & -r_{13} & \cdots \\ 0 & 1 & 0 & \cdots \\ 0 & 0 & \ddots & 0 \end{bmatrix}$$

That is, D_1 has is diagonal with entry every 1, except the first diagonal entry where $d_{11} = \frac{1}{r_{11}}$. Then T_1 is upper-triangular with all diagonal entries 1. Every other entry is 0, except the first row, where $t_{11} = 1$ and $t_{1j} = -r_{1j}$ for $j > 1$. In general, $D_i, T_i \in \mathbb{C}^{n\times n}$ satisfy $R_i = D_i T_i$, where

$$d_{kj} = \begin{cases} \frac{1}{r_{ii}} & \text{if } k = i \text{ and } j = i \\ 1 & \text{if } k = i \text{ and } j \neq i \\ 0 & \text{Otherwise} \end{cases} \qquad t_{kj} = \begin{cases} 1 & \text{if } k = i \\ 0 & \text{if } k \neq i \\ r_{kj} & \text{if } k = i \text{and } j > i \end{cases}$$

In Algorithm 8.1, the D_i matrix corresponds to the step $q_i = v_i / r_{ii}$, which normalizes v_i. The T_i matrix corresponds to the inner loop

$$\begin{aligned} r_{ij} &= q_i^* v_j \\ v_j &= v_j - r_{ij} q_i \end{aligned}$$

where r_{ij} is computed and then multiplied by the outer loop column before it is subtracted.