Computation of Singular Values

Let $A \in \mathbb{C}^{m\times n}$. Consider the SVD

$$A = \underbrace{U}_{m\times m} \underbrace{\Sigma}_{m\times n} \underbrace{V^*}_{n\times n}$$

$$\begin{aligned} AA^* &= (U\Sigma V^*)(U\Sigma V^*)^* \\ &= U\Sigma V^* V\Sigma^* U^* \\ &= U\Sigma\Sigma^* U^* \end{aligned}$$

$\Longrightarrow (AA^*)U = U(\Sigma\Sigma^*)$

$\Longrightarrow (AA^*)u_j = \sigma_j^2 u_j \quad j = 1, \dots, \min\{m,n\}$

$$\begin{aligned} A^*A &= (U\Sigma V^*)^*(U\Sigma V^*) \\ &= V\Sigma^*\Sigma V^* \end{aligned}$$

$\Longrightarrow (A^*A)V = V(\Sigma^*\Sigma)$

$\Longrightarrow (A^*A)v_j = \sigma_j^2 v_j \quad j = 1, \dots, \min\{m,n\}$

Summary

A	A^*A	AA^*
σ – singular value	σ^2 – eigenvalue	σ^2 – eigenvalue
u, v – a pair of left, right singular vectors	v – eigenvector	u – eigenvector

2-stages of Singular Value Computation

① Reduction into bidiagonal form

Form unitary $U \in \mathbb{C}^{m \times m}$, unitary $V \in \mathbb{C}^{n \times n}$ s.t.

$$UAV = B \quad \left(= \begin{bmatrix} \times & \times & & 0 \\ & \times & \ddots & \\ & & \ddots & \times \\ 0 & & & \times \end{bmatrix} \right)$$

is bidiagonal, i.e., $b_{ij} = 0$ whenever $i \neq j$ OR $i \neq j-1$

② QR Algorithm

Form unitary $U_1, \ldots, U_k \in \mathbb{C}^{m \times m}$,
unitary $V_1, \ldots, V_k \in \mathbb{C}^{n \times n}$ s.t.

$$U_k \cdots U_1 B V_1 \cdots V_k = B_k$$

becomes diagonal as $k \to \infty$.

Reduction into bidiagonal form

$$\begin{bmatrix} x & x & x & \cdots & x \\ x & x & x & & x \\ x & x & x & & x \\ \vdots & & & & \vdots \\ x & x & x & & x \end{bmatrix} \xrightarrow[\text{based on } u^{(1)} \text{ from left}]{\text{reflector}} \begin{bmatrix} x & x & x & & x \\ 0 & x & x & & x \\ 0 & x & x & & x \\ \vdots & & & & \\ 0 & x & x & & x \end{bmatrix}$$

$u^{(1)}$ to be modified (TBM)   TBM $[v^{(1)}]^T$

$$\xrightarrow[\text{based on } v^{(1)} \text{ from right}]{\text{reflector}} \begin{bmatrix} x & x & 0 & \cdots & 0 \\ 0 & x & x & & x \\ 0 & x & x & & x \\ \vdots & & & & \\ 0 & x & x & & x \end{bmatrix}$$

kth step

$(k-1)\times(k-1)$ bidiagonal

$$\begin{bmatrix} B_k & C_k \\ 0 & \text{TBM} \end{bmatrix} \xrightarrow[\text{based on } u^{(k)} \text{ from left}]{\text{reflector}} \begin{bmatrix} B_k & C_k \\ 0 & \begin{matrix} x & x & x & x \\ 0 & x & x & x \\ \vdots & & & \\ 0 & x & x & x \end{matrix} \end{bmatrix}$$

$u^{(k)}$ TBM $[v^{(k)}]^T$

$$\xrightarrow[\text{based on } v^{(k)} \text{ from right}]{\text{reflector}} \begin{bmatrix} B_k & C_k \\ 0 & \begin{matrix} x & x & 0 & \cdots & 0 \\ 0 & x & x & & x \\ 0 & x & x & & x \\ \vdots & & & & \\ 0 & x & x & & x \end{matrix} \end{bmatrix}$$

⋮

Pseudocode

```
for j = 1, ..., n-2
    if j < m
        u ← A(j:m, j)
        q ← {u - ||u||_2 e_1} / ||u - ||u||_2 e_1||_2
        A(j:m, j:n) ← A(j:m, j:n) - 2q(q* A(j:m, j:n))
    end
    if j+1 < n
        v^T ← A(j, j+1:n)
        q ← {v - ||v||_2 e_1} / ||v - ||v||_2 e_1||_2
        A(j:m, j+1:n) ← A(j:m, j+1:n) - {A(j:m, j+1:n) 2q} q*
    end
end
B ← A
return
```

\# of flops $\sim 4mn^2 - 4n^3/3$

QR Algorithm

Generates a sequence $\{B_k\}$ s.t.

$$B_0 = B$$

B_{k+1} and B_k are related as follows:

(a) let

$$B_k^* B_k - \sigma_k I = Q_{k+1} R_{k+1}$$

be a QR factorization. (for a shift σ_k)

(b) $$B_k B_k^* - \tilde{\sigma}_k I = P_{k+1} S_{k+1}$$

be another QR factorization. (for another shift $\tilde{\sigma}_k$)

(c) $$B_{k+1} := P_{k+1}^* B_k Q_{k+1}$$

This is a QR algorithm operating simultaneously on B^*B and BB^*, i.e.,

(1) $$B_{k+1}^* B_{k+1} = Q_{k+1}^* \underbrace{B_k^* B_k}_{Q_{k+1} R_{k+1} + \sigma_k I} Q_{k+1}$$

$$= R_{k+1} Q_{k+1} + \sigma_k I$$

(2) $$B_{k+1} B_{k+1}^* = P_{k+1}^* \underbrace{B_k B_k^*}_{P_{k+1} S_{k+1} + \tilde{\sigma}_k I} P_{k+1}$$

$$= S_{k+1} P_{k+1} + \tilde{\sigma}_k I$$

Remarks

① It can be shown that the sequence $\{B_k\}$ is bidiagonal.

② The QR factorizations

$$\underbrace{B_k^* B_k - \sigma_k I}_{\text{Tridiagonal}} = Q_{k+1} R_{k+1}$$

$$\underbrace{B_k B_k^* - \tilde{\sigma}_k I}_{\text{Tridiagonal}} = P_{k+1} S_{k+1}$$

can be computed at a cost of $O(m+n)$ flops.

③ The multiplication

$$P_{k+1}^* B_k Q_{k+1}$$

can be performed at a cost of $O(m+n)$ flops.