# CS475 / CS675
# Lecture 20: July 7, 2016

Bidiagonalization

SVD Image Compression

Reading: [TB] Chapter 31

# Alternative SVD Technique

- Assume $A$ is square, i.e., $m = n$
- Consider the $2n \times 2n$ symmetric matrix:

$$H = \begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix}$$

- Since $A = U\Sigma V^T$, $AV = U\Sigma$, $A^T U = V\Sigma^T = V\Sigma$

then $\begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} V & V \\ U & -U \end{bmatrix} = \begin{bmatrix} A^T U & -A^T U \\ AV & AV \end{bmatrix}$

$$= \begin{bmatrix} V\Sigma & -V\Sigma \\ U\Sigma & U\Sigma \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} V & V \\ U & -U \end{bmatrix}}_{Q} \underbrace{\begin{bmatrix} \Sigma & 0 \\ 0 & -\Sigma \end{bmatrix}}_{\Lambda}$$

# Alternative SVD Technique

- Hence, $HQ = Q\Lambda \rightarrow$ eigendeomposition of $H$
- Algorithm:
  - Compute eigendecomposition of $H$.
  - Set $\sigma_A = |\lambda_H|$
  - Extract $U, V$ from $Q$
- Stable algorithm

# Two-phase SVD

- Idea: First reduce the matrix to bidiagonal form, then diagonalize it.
- Picture:

# Golub-Kahan Bidiagonalization

- Apply Householder reflectors on the left and the right

- $n$ reflectors on the left, $n-2$ on the right
- $flops(bidiag) = 2 \times flops(QR) \approx 4mn^2 - \frac{4}{3}n^3$

# Low-Rank Approximation

- Theorem: $A$ is the sum of $r$ rank-one matrices:

$$A = \sum_{j=1}^{r} \sigma_j U_j V_j^T$$

- Proof:

# Low-Rank Approximation

- Theorem: For any $k$, $0 \leq k \leq r$, define

$$A_k = \sum_{j=1}^{k} \sigma_j U_j V_j^T$$

Then $||A - A_k||_2 = \inf_{rank(B) \leq k} ||A - B||_2 = \sigma_{k+1}$

- Proof: first note that

$$A - A_k = \sum_{j=k+1}^{r} \sigma_j U_j V_j^T = [U_1 \dots U_m] \begin{bmatrix} 0 & & & \\ & \sigma_{k+1} & & \\ & & \ddots & \\ & & & \sigma_r \end{bmatrix} \begin{bmatrix} V_1^T \\ \vdots \\ V_n^T \end{bmatrix}$$

It is the SVD of $A - A_k$

Hence: $||A - A_k||_2 = \sigma_{k+1}$

# Low-Rank Approximation

- Suppose $\exists B$ with $rank(B) \leq k$ such that
$$||A - B||_2 < ||A - A_k||_2 = \sigma_{k+1}$$
- Then $\exists$ $(n - k)$-dim subspace $W$ such that
$$w \in W \Longrightarrow Bw = 0$$
- Note $Aw = (A - B)w$. Then
$$\begin{aligned} ||Aw||_2 &= ||(A - B)w||_2 \\ &\leq ||A - B||_2 ||w||_2 \\ &< \sigma_{k+1} ||w||_2 \end{aligned}$$

# Low-Rank Approximation

- But $\exists$ $(k+1)$-dim subspace $V_{k+1}$ such that $\left||Av|\right| \geq \sigma_{k+1}||v||$
  - E.g., $V_{k+1} = span\{v_1, v_2, \dots, v_{k+1}\}$
  - Note: $Av_j = \sigma_j v_j$,

$$\left||Av_j|\right| = \sigma_j \geq \sigma_{k+1} \left||v_j|\right|$$

- But $\dim(W) + \dim(V_{k+1}) > n$
  $\Longrightarrow$ contradiction

# Low-Rank Approximation

- Notes

1. $A_k = [u_1 \dots u_m] \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_k & \\ & & & 0 \end{bmatrix} \begin{bmatrix} v_1^T \\ \vdots \\ v_n^T \end{bmatrix}$

$$= [u_1 \dots u_m] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix} \begin{bmatrix} v_1^T \\ \vdots \\ v_k^T \end{bmatrix}$$

$$= U_k \Sigma_k V_k^T$$

2. $A_k$ is the best rank-$k$ approximation of $A$.

The error of approximation is $\sigma_{k+1}$ (in $L_2$-norm)

# Application: Image Compression

- An $m \times n$ image can be represented by $m \times n$ matrix $A$ where $A_{ij}$ = pixel value at $(i, j)$

- Compress the image by storing less than $mn$ entries

- Let $A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$, the best rank-$k$ approx of $A$

- Keep the first $k$ singular values and use $A_k$ to approximate $A$; i.e., $A_k$ = compressed image

# Application: Image Compression

- Example: $m = 320, n = 200$
- To store $A_k$, only need to store $u_1, \dots, u_k$ and $\sigma_1 v_1, \dots, \sigma_k v_k$
  - This requires only $(m + n)k$ words
- In contrast, to store $A$ one needs $mn$ words
- Compression ratio: $\frac{(m+n)k}{mn} \approx \frac{k}{123}$

# Application: Image Compression

| k | Relative error $\frac{\sigma_{k+1}}{\sigma_1}$ | Compression rate |
|---|---|---|
| 3 | | |
| 10 | | |
| 20 | | |