Randomized Algorithms & Low-Rank Approximation

Recap

<u>Direct Methods:</u> Compute the core matrix factorizations to ϵ_{mach} back. error:

- => LU
- => QR
- => Eigenvalue decomposition
- => SVD

Typically $\mathcal{O}(m^3)$ for $m \times m$ matrix.

<u>Iterative Methods:</u> Compute an approximate soln to tol. δ in $N(\delta)$ iterations:

- => Arnoldi
- => GMRES
- => Lanczos
- => CG

Eigenvalues

linear systems

Trade off accuracy and computational cost. Usually want fast $x \to Ax$, e.g., $\mathcal{O}(m \log(m))$.

Many applications, but historically these are closely associated w/solving PDE discretizations

Low-rank Structure in Data

Relatively new feature of applied math is sheer volume of data available.

⇒ Need to work with high-dim data sets/matrices.

⇒ Often less <u>explicit</u> structure than, e.g., PDE discretizations.

⇒ Presence of noise and corruption in matrix entries.

How do we deal with high-dim data?

<u>Observation:</u> high-dim data can often be approximated with low-rank matrices.

$$A \approx QB$$

$$\underbrace{m\begin{bmatrix} \quad & \quad \\ \quad & \quad \end{bmatrix}}_{n} \approx \underbrace{m\begin{bmatrix} \quad \\ \quad \end{bmatrix}}_{r}\overset{n}{\begin{bmatrix} \quad & \quad \end{bmatrix}}r$$

column/row spaces are low-dimensional

<u>Idea:</u> Project data onto low-dim subspace and do direct NLA there.

<u>Stage A</u>: Compute ONB Q whose span approximates span(A).

$$\Rightarrow \quad A \approx Q\underbrace{Q^*A}_{B} \quad \text{and} \quad Q^*Q = I$$

<u>Stage B</u>: Compute $B = Q^*A$ and factor B with a direct method

$$\Rightarrow \quad B = U\Sigma V^*, \qquad A \approx (QU)\Sigma V^*$$

How do we find such a Q?

<u>Low-Rank Approximation</u>

Fixed Precision Problem: Given $m\times n$ A and tolerance $\varepsilon > 0$, find $m\times k$ Q s.t.

$$(*) \qquad \|A - Q\underbrace{Q^*A}_{B}\|_2 < \varepsilon \quad \text{and} \quad Q^*Q = I.$$

The SVD of A tells us how to pick Q.

$$\underset{A}{\begin{bmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{bmatrix}} = \underset{U}{\begin{bmatrix} | & & | \\ u_1 & \cdots & u_n \\ | & & | \end{bmatrix}} \underset{\Sigma}{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix}} \underset{V^*}{\begin{bmatrix} - & v_1^* & - \\ & \vdots & \\ - & v_n^* & - \end{bmatrix}}$$

$$= \sum_{j=1}^{n} \sigma_j u_j v_j^* = \sum_{j=1}^{k} \sigma_j u_j v_j^* + \sum_{j=k+1}^{n} \sigma_j u_j v_j^*$$

$$= \underbrace{\begin{bmatrix} | & & | \\ u_1 & \cdots & u_k \\ | & & | \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix} \begin{bmatrix} - & v_1^* & - \\ & \vdots & \\ - & v_k^* & - \end{bmatrix}}_{A_k} + \underbrace{\begin{bmatrix} u_{k+1} & \cdots & u_n \end{bmatrix} \begin{bmatrix} \sigma_{k+1} & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} \begin{bmatrix} - & v_{k+1}^* & - \\ & \vdots & \\ - & v_n^* & - \end{bmatrix}}_{E}$$

$$\boxed{\begin{array}{l} \underline{\text{Thm}} \text{ (Eckart-Young, see L.N.T. lecture 5)} \\ \text{For any } 0 \le k \le r, \text{ we have that} \\ \qquad \|A - A_k\|_2 = \inf_{\operatorname{rank}(X) \le k} \|A - X\|_2 = \sigma_{k+1} \end{array}}$$

"Best rank k approximation of A is given by k-truncated SVD of A."

$\Rightarrow$ To solve $(*)$, choose k s.t. $\sigma_{k+1} < \varepsilon$ and set

$$Q = \begin{bmatrix} u_1 & \cdots & u_k \end{bmatrix} = U_k$$

Then $QQ^*A = A_k$, so

$$\|A - QQ^*A\|_2 = \sigma_{k+1} < \varepsilon. \quad \checkmark$$

Randomized "Sketch"

Truncated SVD of A is best low-rank approx, but too expensive to compute.

Idea: Try to approximate $\mathrm{span}(U_k)$, the dominant singular vectors of A, with a single power iteration.

Intuition 1: Suppose A has exactly rank k, so that $A = A_k$ and $\sigma_k > 0$.

$$\underbrace{\begin{bmatrix} | & & | \\ w_1 & \cdots & w_k \\ | & & | \end{bmatrix}}_{W} = \underbrace{\begin{bmatrix} | & & | \\ u_1 & \cdots & u_k \\ | & & | \end{bmatrix}}_{U_k} \underbrace{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix}}_{\Sigma_k} \underbrace{\begin{bmatrix} - & v_1^* & - \\ & \vdots & \\ - & v_k^* & - \end{bmatrix}}_{V_k} \underbrace{\begin{bmatrix} | & & | \\ x_1 & \cdots & x_k \\ | & & | \end{bmatrix}}_{X}$$

W: random samples from $\mathrm{span}\{u_1, \ldots, u_k\}$

X: random vectors

W has lin. indep. columns as long as

$V_k^* X$ has full rank, $\mathrm{rank}(V_k^* X) = k$.

$\Rightarrow$ for i.i.d. normal entries of X, this happens almost surely.

To get a ONB, we orthonormalize

$\Rightarrow$ compute $W = QR$

$\therefore \quad \|A - QQ^* A\|_2 = \sigma_{k+1} = 0 \quad$ a.s.

<u>Intuition 2</u>: Now, let $A = A_k + E$ as before so that $\|E\|_2 = \sigma_{k+1} < \varepsilon$.

$$\tilde{w}_j = A x_j = A_k x_j + E x_j$$

$$\underbrace{\tilde{W}}_{} = \underset{\substack{\uparrow \\ \text{basis for} \\ \text{span}(A_k) \\ \text{as before}}}{W} + \underset{\substack{\uparrow \\ \text{small} \\ \text{perturbation}}}{EX}$$

The span of $\tilde{W}$ is a good approx to W with high probability as long as we <u>oversample</u>, taking x_j for $1 \le j \le k + \underset{\substack{\uparrow \\ \text{oversampling} \\ \text{param.}}}{p}$