Announcements

HW1

SRFT → see forum follow-up

Goals:

- Approx SVD in $O(N)$ time
- RRQR
- Interpolative decomp.

Review:

- $\overset{R}{\vee}$range finder
  ↳ adaptive
- reconstructing the SVD

$$U \Sigma V^T = \underbrace{Q^T A}$$

Computing $Q^T A^{k \times N}$ is $O(N^2 k)$

Errors in Random Approximations

If we use the randomized range finder, how close do we get to the optimal answer?

Theorem

For an $m \times n$ matrix A, a target rank $k \geqslant 2$ and an oversampling parameter $p \geqslant 2$ with $k + p \leqslant \min(m, n)$, with probability $1 - 6 \cdot p^{-p}$,

$$\left|A - QQ^T A\right|_2 \leqslant \left(1 + 11\sqrt{k+p}\sqrt{\min(m,n)}\right)\sigma_{k+1}.$$

(given a few more very mild assumptions on p)

[Halko/Tropp/Martinsson '10, 10.3]

Message: We can *probably* (!) get away with oversampling parameters as small as $p = 5$.

A-posteriori and Adaptivity

The result on the previous slide was *a-priori*. Once we're done, can we find out 'how well it turned out'?

$$\|E\| = \|A - QQ^T A\|$$

$$\|E\|_2 \approx \frac{\|A\vec{\omega}\|_2}{\|\vec{\omega}\|_2} \qquad \omega \in \mathbb{R}^n \text{ Gaussian iid}$$

Adaptive Range Finding: Algorithm

(see above)

Outline

Rank-revealing/pivoted QR

$PA = LU$

Sometimes the SVD is too *good* (aka expensive)–we may need less accuracy/weaker promises, for a significant decrease in cost.

$$A\Pi = QR = Q\begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix}$$

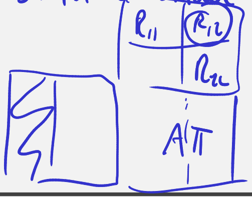
$R_{11} \in \mathbb{R}^{k \times k}$

$\|R_{22}\|_2$ hopefully small

$\|R_{12}\|$? no reason that should be small

Q orthogonal

Π is a permutation



Q

Using RRQR for LRA

- $\sigma_{k+1} \le \|R_{22}\|_2$
- To precision $\|R_{22}\|_2$, A has at most num. rank k

  (Golub/van Loan, ch. 5)

Also suitable for LRA, but:

need heuristic to stop half-way

Does not qualify if full matrices are required.

Interpolative Decomposition (ID): Definition

Would be helpful to know *columns of A* that contribute 'the most' to the rank.
(orthogonal transformation like in QR 'muddies the waters')

If A has rank (k), then $\vec{j}$ is a k-long index v.

$$A_{m\times n} = (A_{(:,\vec{j})})_{m\times k} P_{k\times n}$$

- k columns of P contain a single entry of 1
- "P is well-conditioned"
  all entries are bounded by 2.

ID: Computation

Assume $\operatorname{rank}(A) = k$

How do we construct this (from RRQR): (short/fat case)

$$\underset{m\times n}{A}\ \underset{n\times n}{\Pi} = \underset{m\times k}{Q}\underbrace{(\overbrace{R_{11}}^{k\times k}\ R_{12})}_{k\times n} \qquad \underset{m\times k}{B} = \underset{m\times k}{Q}\ \underset{k\times k}{R_{11}}$$

Q: What is P, in terms of the RRQR?

$$P = [\mathrm{Id} \quad R_{11}^{-1} R_{12}]\ \Pi^T$$

$$BP = QR_{11}(\mathrm{Id} \quad R_{11}^{-1}R_{12}]\ \Pi^T$$

$$= Q[R_{11} \quad R_{12}]\ \Pi^T$$

$$BP\Pi = Q[R_{11} \quad R_{12}]$$

$$A\Pi = Q[R_{11} \quad R_{12}]$$

ID Q vs ID A

What does row selection mean for the LRA?

$$A \approx Q Q^T A$$

Suppose I run a <u>row</u> ID on Q: $\quad Q \approx P\, Q_{[j,:]}$

$$A \approx P\, Q_{[j]}\, Q^T A$$

$$A_{[j]} \approx \overset{\text{Id}}{\cancel{P_{[j]}}}\; Q_{[j]})\; Q^T A$$

[Martinsson, Rokhlin, Tygert '06]

ID: Remarks

Slight tradeoff here: what?

How would we use the ID in the context of the range finder?

Demo: Interpolative Decomposition