DS 298: Work Assignment - 3

Due April 11, 2024

The goal of this work is to analyze the efficiency of the three randomized linear-solve algorithms (1) Kaczmarz (2) CD-Least Square and (3) CD-SPD, in solving symmetric-positive-definite (SPD) linear systems.

Let $Ax = b$ be the linear system considered where a matrix A is generated using algorithm-1, and b is randomly generated as Ax with the entries of the random x from $\mathcal{N}(0, 1)$ for any matrix A. The expected computing effort required to reach a particular relative error $\|x - x_k\|/\|x\|$ is of interest here, and can be estimated using the number of iterations required and the number of scalar multiplications involved in each iteration. Let the starting $x_0 = \hat{0}$ in all cases implying a relative error of 1 to start with in the first iteration. The expected computing effort to reach a relative error ≤ 0.1 is to be evaluated by averaging over the random vectors b for each $n \times n$ matrix A generated. Use values of n (x-axis in plots) as 100, 200, 400, 800, 1600. Average the computing effort over sufficient number of vectors b for each data point in the plot, and you are allowed to stop a trial if it does not converge in $10n$ iterations and include the expended computing effort in the estimates. This semi-log plot (with y-axis in log scale) in comparing the computing effort of the three algorithms, is to be replicated for the two matrix classes I and II.

The trial SPD matrices A are to be generated using the eigenvalue decomposition $Q\Lambda Q^T$ as shown in algorithm-1. The eigenvalues given by Λ_{kk} should be fixed as i) $2e^{-\frac{10(k-1)}{n}}$ for Class-I matrices, and as ii) $1 + \frac{k}{n}$ for Class-II matrices. Also, ensure that each iteration of the three linear-solve algorithms implemented in code actually scale as $\mathcal{O}(n)$; submit example plots of the wall-time taken with the increasing number of iterations, in an appendix, to confirm the same. Note that the probabilities of sampling the k^{th} column (or row) in the algorithms are given by

$$p_k = \frac{\|A^{(k)}\|^2}{\sum_{k=1}^{n} \|A^{(k)}\|^2}$$

.

Note : Submit the responses with the plots, and the codes, as separate files all zipped into a single folder identified by your name in full, to *nagagn@iisc.ac.in* (or upload in MS-Teams as specified in its $DS298$ channel).

Algorithm 1 : Generation of a trial matrix in a SPD class

Inputs : $\Lambda \in \mathcal{R}^{n\times n}$; diagonal matrix of eigenvalues.
Outputs : $A \in \mathcal{R}^{n\times n}$; trial matrix.
Initialize : $M \longleftarrow rand[n, n]$; Initialize a random matrix.
Generate eigenvectors : $QR \longleftarrow M$; QR factorization of the random matrix.
Build trial matrices : $A \longleftarrow Q\Lambda Q^T$; generate test matrix of a given eigenvalue distribution.