

Stat243: Problem Set 7, Due Wednesday November 17

November 8, 2021

This covers Units 9 and 10.

It's due **as PDF submitted to Gradescope** and submitted via GitHub at 10 am on Nov. 17.

Comments:

1. The formatting requirements are the same as previous problem sets.
2. Please note my comments in the syllabus about when to ask for help and about working together. In particular, **please give the names of any other students that you worked with on the problem set and indicate in comments any ideas or code you borrowed from another student.**

Problems

1. Compare the speed of $x = A^{-1}b$ using: (i) `solve(A) %*%b`, (ii) `solve(A, b)`, and (iii) Cholesky decomposition followed by solving triangular systems. To ensure that A is invertible, you can construct a matrix A as $A = W^T W$ where the elements of the $n \times n$ matrix W are generated independently using `rnorm()`. Take $n = 5000$. Note, if your R installation is not using a fast BLAS package, all three of these approaches will likely take a lot longer than if you are using a fast BLAS (e.g., on the SCF). See Section 6.1 of Unit 10 and/or Section 4.1.2 of Unit 8.
 - (a) **Using a single thread**, how do the timing and relative ordering amongst methods compare to the order of computations we discussed in class and the notes? Note that if one works out the complexity of the full inversion using the LU decomposition, it is $4n^3/3$.
 - (b) Are the results for b the same numerically for methods (ii) and (iii) (up to machine precision)? Comment on how many digits in the elements of b agree, and relate this to the condition number of the calculation. (Note: you may want to compute the condition number on an SCF machine as it can take a few minutes to do the eigendecomposition for a matrix of this size.)
2. The following calculation arises in solving a least squares regression problem where the coefficients are subject to an equality constraint, in particular, we want to minimize $(Y - X\beta)^T(Y - X\beta)$ with respect to β subject to the m constraints $A\beta = b$ for an m by p matrix A . (Each row of A represents a constraint that that linear combination of β equals the corresponding element of b .) Solving this problem is a form of optimization called quadratic programming. Some derivation using the Lagrange multiplier approach gives the following solution:

$$\hat{\beta} = C^{-1}d + C^{-1}A^T(AC^{-1}A^T)^{-1}(-AC^{-1}d + b)$$

where $C = X^T X$ and $d = X^T Y$. X is n by p .

- (a) Describe how you would implement this in pseudo-code, taking account of the principles discussed in class in terms of matrix inverses and factorizations
 - (b) Write an R function to efficiently compute $\hat{\beta}$, taking account of the principles discussed in class in terms of matrix inverses and factorizations. Note: you can use any of R's matrix manipulation functions that you want - I'm not expecting you to code up any algorithms from scratch.
Note: in reality a very efficient solution is only important when the number of regression coefficients, p , is large.
3. Details of the Cholesky decomposition presented in Unit 10. Work out the operation count (total number of multiplications plus divisions) for the Cholesky decomposition, including the constant c , not just the order, for terms involving n^3 or n^2 (e.g., $5n^3/2 + 8n^2$, not $O(n^3)$). You can ignore the square root and any additions/subtractions. You can ignore pivoting for the purpose of this problem. Remember not to count any steps that involve multiplying by 0 or 1. Compare your result to that given in the notes.
4. **(Extra credit)** In class we saw that the condition number when solving a system of equations, $Ax = b$, is the ratio of the absolute values of the largest and smallest magnitude eigenvalues of A . Show that $\|A\|_2$ (i.e., the matrix norm induced by the usual L2 vector norm; see Section 1.6 of Unit 10) is the largest of the absolute values of the eigenvalues of A for symmetric A . To do so, find the following quantity,

$$\|A\|_2 = \sup_{z: \|z\|_2=1} \sqrt{(Az)^\top Az}.$$

If you're not familiar with the notion of the supremum (the *sup* here), just think of it as the maximum. It accounts for situations such as trying to find the maximum of the numbers in the open interval (0,1). The max is undefined in this case since there is always a number closer to 1 than any number you choose, but the *sup* in this case is 1.

Hints: when you get to having the quantity $\Gamma^\top z$ for orthogonal Γ , set $y = \Gamma^\top z$ and show that if $\|z\|_2 = 1$ then $\|y\|_2 = 1$. Finally, if you have the quantity $y^\top Dy$, think about how this can be rewritten given the form of D and think intuitively about how to maximize it if $\|y\|_2 = 1$.