

STAT 513/413, ASSIGNMENT 2

1. Show (theoretically, so also some proof required) how the determinant of a matrix can be computed via the eigenvalue decomposition. For simplicity, consider only diagonalizable matrices, matrices that have all eigenvalues real. In statistical context, we are often fine working only with symmetric matrix, which are all diagonalizable. Can the determinant be compute also via singular value decomposition? Explain why yes or no.

2. Devise how the Cholesky decomposition of a symmetric positive definite matrix can be obtained via eigenvalue and QR decompositions. Some proofs required: in particular, demonstrate that the method will work for *every* positive definite symmetric matrix. Finally, check (or better word: test) your proposed method also computationally: implement your method and verify that it works, on one or two examples. (You may also compare it to the result of `chol()` function in R, evaluated for your test examples, but that is not required.)

3. Consider the distribution with a density

$$f(x) = \begin{cases} \frac{3}{4}x(2-x) & \text{for } x \in [0, 2] \\ 0 & \text{otherwise} \end{cases}$$

(a) Construct a function that returns random numbers from this distribution using acceptance/rejection method and implement it. (Regarding theory, there is almost nothing to do here.

(b) Construct another function that returns random numbers from this distribution, but now using inversion method. Work out the necessary theory first.

(c) Implement the function designed under (b), and compare the results of both functions with the help of `qqplot`.

Note: the solution of (c) may require some algebra that is well-known, but is not routinely used, and may become a bit tedious if done in an uninspired way. Programming arcane algebraic formulas is not an objective here: feel free to use some convenient R function or package for the algebra here if necessary (packages are thus allowed for this step). Use Google and your judgment here: the package may offer various solutions, in which case you may need to think probabilistically again to decide which are the right ones - which, on the other hand *is* an objective here.

4. A **run** of length k is a sequence of k adjacent same symbols. Develop an algorithm counting the number of runs of length k in a 0-1 sequence of length n , and use it to

(a) estimate the probability that at least one run of length k will appear in such a sequence;

(b) estimate the expected number of runs of length k in such a sequence.

Report results for $n=50$ and 100 , and for $k=2,3,4,5,6,7$. Try different number of repetitions, to have some reassurance that your probability estimates are good for two significant digits.

Read carefully: the text above calls for the algorithm that counts the number of runs of length k . Those are sequences of k adjacent same symbols. Nothing else; in particular, it is nowhere written there that they should be counted as “non-overlapping” or “maximal” runs. To illustrate it on an example, the sequence

0 1 1 0 0 0 0 1 1 1

has 6 runs of length 2, 3 of length 3, 1 of length 4, and 0 for any $k \geq 5$.

Due Monday, February 8, 2021