Homework 3-2 (50 pts)

1 Use Matlab for Cholesky Factorization

In this homework, you need to implement Cholesky factorization in outer product form using **MATLAB**. (You may use 217's machines.) If you do not have MATLAB, use **Octave** instead. In the outer product form, your procedure recursively calculates a smaller matrix:

$$\begin{bmatrix} \alpha & \mathbf{v}^T \\ \mathbf{v} & \mathbf{B} \end{bmatrix} = \begin{bmatrix} \sqrt{\alpha} & \mathbf{0}^T \\ \frac{\mathbf{v}}{\sqrt{\alpha}} & \mathbf{I} \end{bmatrix} \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{B} - \frac{\mathbf{v}\mathbf{v}^T}{\alpha} \end{bmatrix} \begin{bmatrix} \sqrt{\alpha} & \frac{\mathbf{v}^T}{\sqrt{\alpha}} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \sqrt{\alpha} & \mathbf{0}^T \\ \frac{\mathbf{v}}{\sqrt{\alpha}} & \mathbf{L} \end{bmatrix} \begin{bmatrix} \sqrt{\alpha} & \frac{\mathbf{v}^T}{\sqrt{\alpha}} \\ \mathbf{0} & \mathbf{L}^T \end{bmatrix},$$

where

$$\mathbf{B} - \frac{\mathbf{v}\mathbf{v}^T}{\alpha} = \mathbf{L}\mathbf{L}^T \tag{1}$$

When updating the matrix

$$\mathbf{B} - \frac{\mathbf{v}\mathbf{v}^T}{\alpha},$$

consider these three implementations:

- No loop (vectorized): Use matrix/vector operations provided by Matlab to do
$$\mathbf{B} = \mathbf{B} - \frac{\mathbf{v}\mathbf{v}^T}{\alpha}.$$
- One-level loop: Update each column of $\mathbf{B}$ in a for loop. That is,
$$\mathbf{B}_{:,j} = \mathbf{B}_{:,j} - \mathbf{v}\frac{v_j}{\alpha}$$
for each j.
- Two-level loop: Update each element of $\mathbf{B}$ in two for loops. That is,
$$\mathbf{B}_{i,j} = \mathbf{B}_{i,j} - v_i\frac{v_j}{\alpha}$$
for each i and j.

Requirements

(i) **Implement** the three above-mentioned versions and **include the code** for all experiments in your report.

(ii) Generate some $n \times n$ symmetric **positive definite** matrix and then verify your program results with the built-in function "chol". Explain how you generated the positive definite matrices.

(iii) Compare the efficiency of your variants under different n, where we suggest you at least include $\{10, 100, 1000, 4000\}$ in the range of n, and measure the running time (elasped time). Depending on the power of your computer, the range of n can be smaller or larger. For timing, you can check

```
>> help toc
```

(iv) Explain the results of (iii). Hint: Read about "vectorization" for MATLAB/Octave.

(v) (**Extra** 10 pts) If you used MATLAB, do (iii) again in Octave and discuss whether the same thing happens. If you used Octave, do the same thing with MATLAB instead. The extra points are added to the total score of HW3 (so it can be added to HW3-1), but the total score of HW3 will not exceed 100 pts.