2**018 Scientific Computing, Computer Project 6**

Jacobi method for computing eigenvalues, 110 points

1. Generate an N by N symmetric matrix as follows:

*void gen\_mtx(double \*\*A, int n){*

*srand(0);*

*for(i=0;i<n;i++)*

*for(j=i;j<n;j++) A[i][j] = A[j][i] = (double)(rand()%20)+1.0;*

*}*

1. Implement the Jacobi method to compute the eigenvalues and eigenvectors of matrix ***A***.
2. Let N = 4. Compute the eigenvalues and eigenvectors. Then print out the following results: (50%)
   1. Matrix ***A*** after the Jacobi procedure converges.
   2. The eigenvalues and the eigenvectors
   3. The norms of , where are eigenvectors and eigenvalues.
   4. Compute  and print the results.
   5. Number of iterations (Given’s rotations)
3. Let N = 25. Compute the following value after completing each iteration (Given’s rotation), assuming k is the index of the iteration:

.

1. Assume *M* iterations have been carried out. Draw a figure to show the variation of *offDiag[k], k=0,1,2,…M-1*. Please print out the eigenvalues too. (20%)
2. Try N= 3, 4, 5, …, 20. Collect the numbers of iterations, *M*, performed in these cases. Draw a figure to show the relation between N and *M*. (20%)
3. It has been proved that Jacobi method enjoys a linear converge rate. However, someone claimed that Jacobi method converges quadratically after some iterations. Based on the test results of 3, 4 and 5, is this claim true? (10%)
4. Implement the max-heap data structure to store the off-diagonal entries such that the max-off-diagonal entry can be found in steps. Compare and show the CPU times of the original implementation and the new one.