

Questions

1. Looking at the graphs provided in the zip file you can see that the offsets start out very close to the theoretical bound but then quickly drop away from the bound. The offsets generated by the program approach 0 much faster than the theoretical bound does.
2. The sorting step of Jacobi Iteration is actually very important. It is so important such that when it is removed from the process the iteration does not finish on my computer. The program will hang because the offsets never reach the termination state. So I could not actually produce graphical results

Implementation

For this project I used Java as my language of choice. Java is my most comfortable language so the choice was very easy for me to make. From my previous project I knew about a linear algebra library known as JAMA. JAMA was a major part of this project as it allowed me to find the Eigen values and vectors necessary to compute a Givens rotation.

Main method

The main method is where all pieces of the program are put together in order to find the offsets and to diagonalize the generated matrices. The main method performs 10 runs of the Jacobi Algorithm on 10 different randomly generated 5x5 symmetric matrices. After the Jacobi algorithm is finished running on each matrix the main method creates a file that contains the diagonalized matrix and a listing of the offsets.

Iterate method

This method handles a single iteration step of the Jacobi algorithm, and is called by the main method until the most recent offset is less than 10^{-9} . First, this method finds the location of the largest off diagonal entry of the matrix. Then it creates a Givens rotation using the methods we learned in class. Lastly, it calculates and returns the new matrix given by GBG^T .

findMax method

This method is called in the iterate method and is used to find the location of the largest off diagonal entry. This method is a fairly standard brute force search of the off diagonal entries. After searching through the upper triangle portion of the matrix the method will return an array containing the values of the indexes of the max value.

offCalc method

This method is called in the main method and is used to calculate the most recent offset. This method works by iterating over the entire matrix and summing the square of the off diagonal values. The value returned is the sum described previously

genMatrix method

This method is called in the main method and is used to generate the random symmetric matrix. This method makes use of JAMA's random matrix function in order to get a normal distribution of values. The values generated are multiplied by 10 in order to make the matrix larger without changing the distribution. Next this method turns the matrix into a symmetrical one by setting the values of the lower entries to the values of their respective upper entries. The method then returns the newly generated matrix.