Jacobi Algorithm Project

By Sneha Ganesh

11/29/2012

Professor Lounici – MATH 2605

DEFINING THE PROBLEM – WHAT IS THE JACOBI ALGORITHM?

The Jacobi algorithm is an approximation method which find the eigenvalues of a matrix by creating similar matrices until the off-diagonal matrices are as small as possible (depending on the precision you want). In the program I created, this is calculated with random 5x5 matrices hence 5 eigenvalues exist.

I decided to create this program in an applet format so it's clear to see the original matrix, the eigenvalues and the graph all in one location.

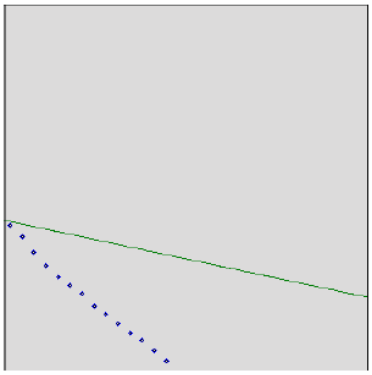
THE APPLET

This program I created runs the Jacobi algorithm on a randomly generated 5x5 matrix until the following statement:

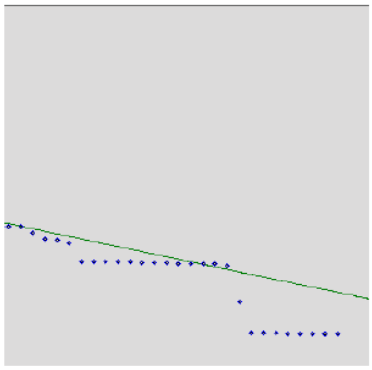
$$Off(B) = \sum_{i \neq j} B^2{}_{i,j} \leq 10^{-9}$$

The applet graphs this general decrease in sequence of numbers and computes the time taken and the number of steps taken to complete the algorithm.

EXAMPLE GRAPH GENERATED



Graph generated from a sorted run



Graph generated from an unsorted run

THE RESULTS

After running the algorithm multiple times, I came up with the following conclusion about each method: Sorted and Unsorted.

The sorted algorithm takes fewer steps than the unsorted algorithm but more time as it has to iterate to find the largest value. This concludes that the unsorted method is not constrained to the same theoretical bounds as the sorted run as it would have a much slower worst-case scenario.

To optimize the algorithm, there should be a way to integrate the sorted iterations and the unsorted runtime to create an enhanced algorithm.

COMPILATION OF 15 RUNS

The following table shows the data generated from running the sorted run 15 times and the unsorted run 15 times. This data clearly emphasizes the conclusion made in the Results section. Sorted Run Unsorted Run

	Sorted Run		Unsorted Run	
	Number of Steps	**Total Run Time (seconds)**	**Number of Steps**	**Total Run Time (seconds)**
	26	0.00897	36	0.00415
	25	0.0164	35	0.0141
	28	0.0191	36	0.00976
	26	0.0188	35	0.0113
	25	0.0202	35	0.0139
	27	0.0137	36	0.0143
	26	0.0302	35	0.00433
	25	0.00664	41	0.0177
	27	0.0171	41	0.015
	25	0.0315	31	0.0133
	25	0.00345	32	0.0126
	27	0.0145	36	0.0312
	27	0.0075	38	0.0136
	26	0.00806	35	0.0125
	26	0.0125	35	0.0122
	24	0.0105	36	0.005
Average	**25.95**	**0.0149**	**35.81**	**0.0128**