Ay190 – Worksheet 09 - LSE
Chatarin (Mee) Wong-u-railertkun
Date: February 11, 2014

1 Is the system solvable?

We read the data of A matrix and **b** vector. In order of the system to be solvable, $det(A) \neq 0$, $\mathbf{b} \neq \mathbf{0}$, and, the size of A and b must be compatible. First, after importing data into `Numpy` array, I make sure that b is not equal to zero vector. In table 1, we can see that all the linear equation systems are solvable. (Even though python spits out Overflow while it tries to calculate some determinant, it's not a problem, yet, since Overflow means that the determinant is not zero.)

i	size(A)	size (b)	det(A)
1	(10, 10)	(10,)	0.0107420716256
2	(100, 100)	(100,)	-3.083846412324e+33
3	(200, 200)	(200,)	-1.51714199214e+98
4	(1000, 1000)	(1000,)	Overflow (-inf)
5	(2000, 2000)	(2000,)	Overflow (+inf)

Table 1: The table shows size of A matrix and b vector, and det(A). Since, for all the systems, the size of A and b are compatible, and det(A) are not equal to zero, they are all solvable.

2 Gaussian Elimination

I use `scipy.linalg.lu` to perform the Gaussian elimination. Then, I wrote a function to solve the linear system. In order to test the function, I randomly generated 3x3 A matrix and vector x of length 3. Multiply both of them to find b, and input A and b into the linear equations solver function. Then, by comparing the error between the generated x and the result from the solver function, we can see how good the Gaussian Elimination method is. Table 2 shows absolute error being very close to zero.

Index	Absolute error
1	1.11022302e-16
2	3.33066907e-16
3	3.33066907e-16

Table 2: The table shows absolute error between the analytical answer of x and the x from linear equation solver using Gaussian Elimination.

3 Computing Time

Comparing time taken to solve linear equations of different size, with the Gaussian Elimination method and `Numpy.linalg.solve`, which uses LAPACK routine _gesv

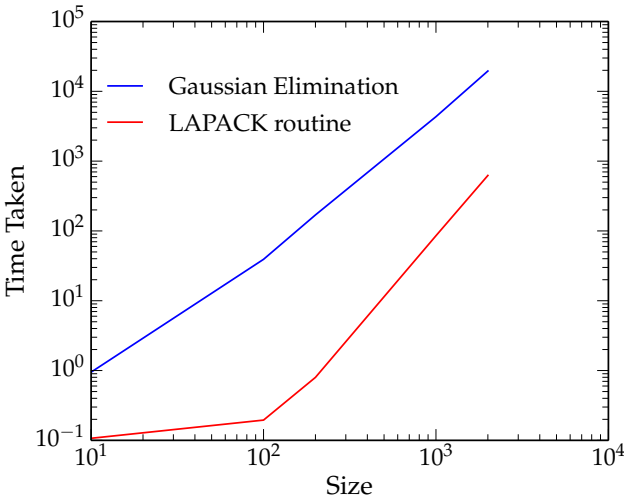

Figure 1: Time used to solve linear equations of different sizes with two methods.