Ay190 – Worksheet 9

Xiangcheng Ma

Date: February 4, 2014

Linear System

(1) Using `np.loadtxt` to read A and **b**:

```
A=np.loadtxt("LSEi_m.dat")
b=np.loadtxt("LSEi_bvec.dat")
```

The size of these systems are 10, 100, 200, 1000 and 2000 from $i = 1$ to 5. Using `SciPy` routine `numpy.linalg.slogdet` to calculate the logarithmic of the determinant of A to determine whether these systems are solvable. The result is all of them can be solved.

(2) I provide my own Gauss elimination routine `gaussian.py`. Inside it, I add a few sentence in order to jump out the routine and claim an error if the systems is not solvable. The routine has been widly tested using a series of different 3×3 systems, including some of extreme cases. The solutions are solved as "`LSEi_soln.dat`". The time consumption is listed in Table 1 and plotted in Figure 1. In general, it follows an $O(N^3)$ complexity.

(3) I use `scipy.linalg.lu_solve` routine, which is based on LU decomposition method. The time comsumption is listed in Table 1 and plotted in Figure 1 for a reference. This routine is much faster than Gaussian elimination. It looks like a complexity of $O(N^n)$ ($n < 3$) for smaller N, while incease to $O(N^3)$ for large N. However, this may result from system flutuations and therefore needs more tests.

i	Size N	Gauss elimination time (s)	LU decomposition time (s)
1	10	0.00195908546448	0.000194072723389
2	100	0.15620803833	0.000787019729614
3	200	0.619211912155	0.00124096870422
4	1000	18.398884058	0.038074016571
5	2000	80.3887019157	0.187728881836

Table 1: Time consumption of linear solvers.

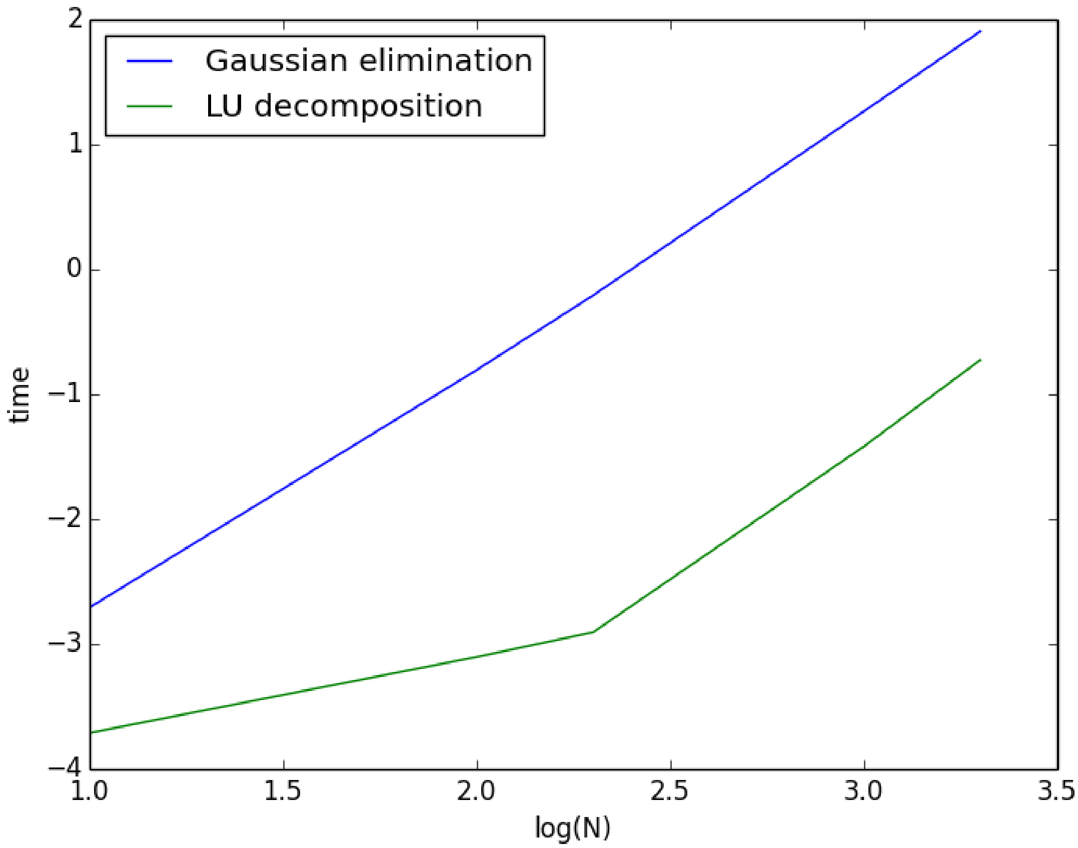

Figure 1: Linear System Solver Time Consumption