Matrix Effiencies for MCMC

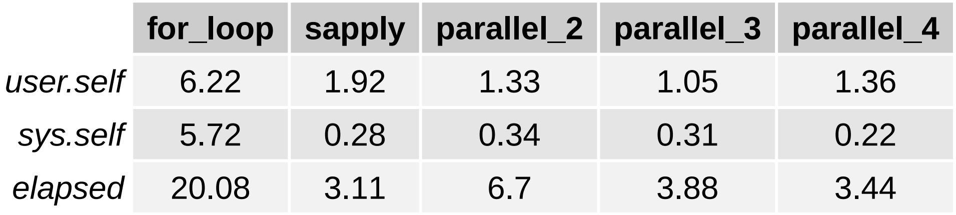
May 20, 2020

Introduction

Indicative report comparing R matrix efficiencies. The initial investigation is on R native solutions including techniques and support libraries for efficiencies. Elementwise function change to 2 dimiensional matrix of 1000 x 1000 matrix with:

- stardard looping through matrix
- sapply
- parrallel (n clusters)

To measure perfomance proc.time function was applied to determines how much real and CPU time (in seconds) the currently running R process has already taken. The user time is the CPU time charged for the execution of user instructions of the calling process. The system time is the CPU time charged for execution by the system on behalf of the calling process. 1



	for_loop	sapply	parallel_2	parallel_3	parallel_4
user.self	6.22	1.92	1.33	1.05	1.36
sys.self	5.72	0.28	0.34	0.31	0.22
elapsed	20.08	3.11	6.7	3.88	3.44

Results

sapply performce show the lowest elapse time on a 1000 x 1000 matrix, this is likely due to the overhead of running jobs in parallel. If the jobs you fire at the worker nodes take a significant amount of time then it is likely parallelization will improve overall performance. If individual jobs take only milliseconds, the overhead of constantly firing off jobs will deteriorate overall performance. Further optimising is possible by dividing the work over the nodes in such a way that the jobs are sufficiently long, say at least a few second. Parrallelisation advantages are shown when runing models simultaneously that individual will run for hours.

Run Time Plots

Computational plots:

Possible Improvements to R code

If R code isn't fast enough due to bottle necks the following mabye possible.

1. Rewrite functions in C/C++, the bottlenecks that can be address are:

- Loops that cant be easily vectorised because subsequent iterations depend on previous ones.
- Recursive functions
- Problems that require advanced data structures

2. Frameworks

- tensorflow (utilisig monder architecturd -GPU)
- mapreduce, Hadoop (clusters for big data)

3. Other improved library languagues

- GNU GSL
- Stan