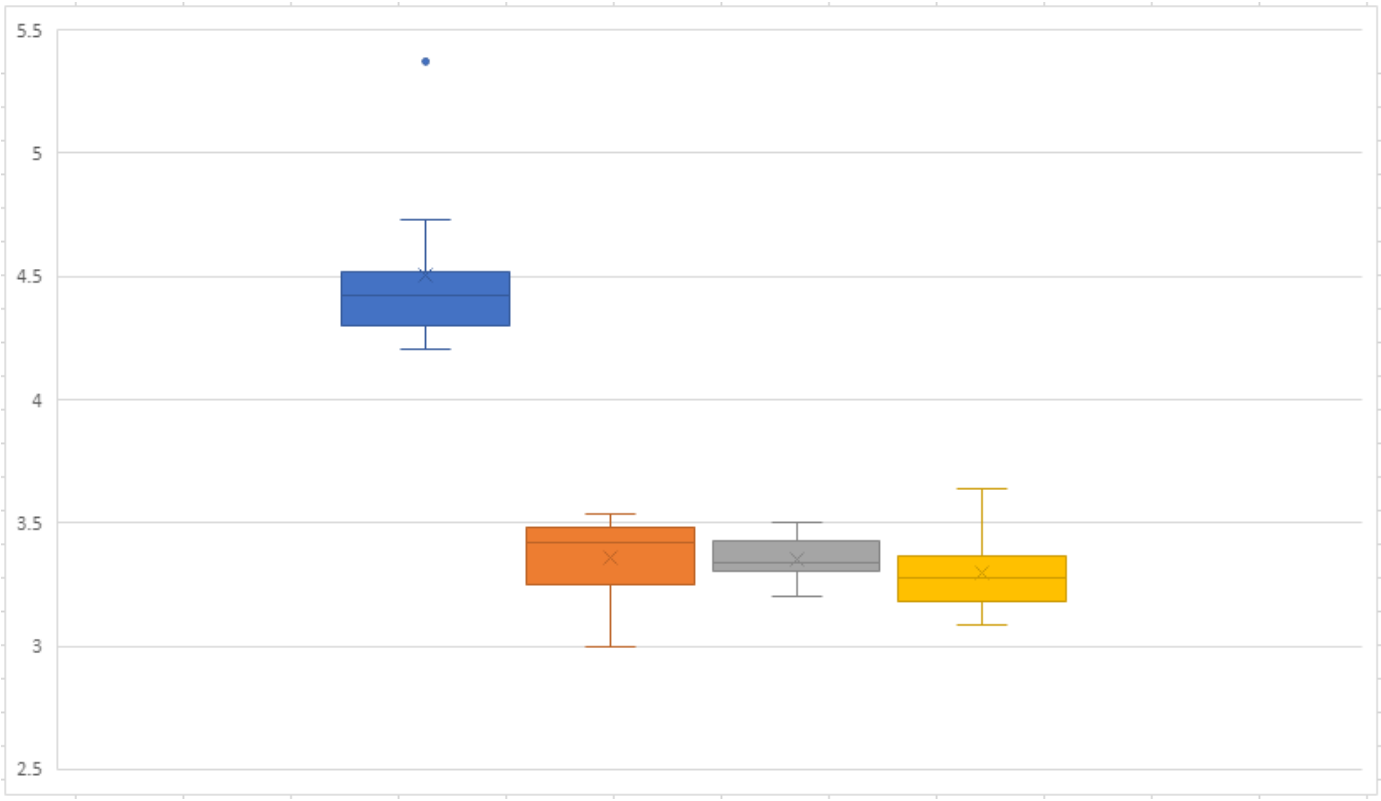
Data Parallelism Results

My expectations:

- Most efficient is:
- Least efficient is having one (1) thread doing all the work. I already knew having only one thread doing all the work would be the least efficient and would results in high run-time
- In terms of my expectations in regards to my program - I would have thought that having at least four (4) threads doing work in data-parallel would be a bit more efficient and quicker in comparison to the other three runs.

Results:



Assignment 1 - Data Parallelism

Thread(s) 1	Thread(s) 2	Thread(s) 3	Thread(s) 4
4.4706	2.9985	3.4507	3.6383
5.3723	3.2254	3.1998	3.2906
4.4768	3.3956	3.3012	3.5276
4.3817	3.1419	3.3193	3.3902
4.5341	3.5390	3.3646	3.0874
4.7305	3.5339	3.5005	3.1649
4.3787	3.4482	3.2674	3.2199
4.2522	3.3310	3.4614	3.2999
4.2040	3.4933	3.3576	3.2599
4.2703	3.4527	3.3133	3.0903

	Series 1	Series 2	Series 3	Series 4
Minimum	4.2040	2.9985	3.1998	3.0874
Q1	4.2974	3.2518	3.3042	3.1787
Median	4.4262	3.4219	3.3385	3.2753
Q3	4.5198	3.4832	3.4292	3.3676
Maximum	5.3723	3.5390	3.5005	3.6383
Mean	4.5071	3.3560	3.3536	3.2969
Range	1.1683	0.5405	0.3007	0.5509

Task Parallelism Results

My expectations:

- I assumed that with Task_parallelism it would be quickly done with a large number of iterations and grid-size. Turns out it took almost half a minute to produce results

Results:

Task Parallelism
24.6047
18.9338
17.3442
18.2299
17.1801
17.1582
18.3327
19.1317
16.7626
16.5072