Name: John Reddy Peasari

Assignment: Week04

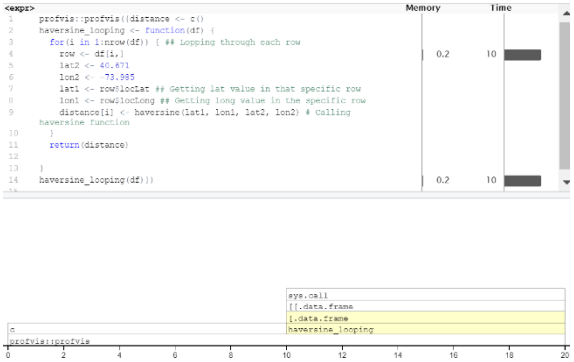
a) Tabulate the execution times of each of the individual approaches for computing distance in Python (i.e., run the shared code on your computer, note the times, and tabulate them).

Method type	%%timeit	cProfile
apply()	45.3 ns ± 0.923 ns per loop (mean ± std. dev. of 7 runs, 10000000 loops each)	5594 function calls (5421 primitive calls) in 0.014 seconds
for loop	46 ns ± 2.44 ns per loop (mean ± std. dev. of 7 runs, 10000000 loops each)	32540 function calls (32163 primitive calls) in 0.035 seconds
Numpy	107 µs ± 6.16 µs per loop (mean ± std. dev. of 7 runs, 10000 loops each)	202 function calls (196 primitive calls) in 0.001 seconds
vectorized	1.71 ms ± 57.6 µs per loop (mean ± std. dev. of 7 runs, 1000 loops each)	5352 function calls (5293 primitive calls) in 0.008 seconds
iterrows()	3.54 ms ± 349µs per loop (mean ± std. dev. of 7 runs, 100 loops each)	129 function calls (123 primitive calls) in 0.000 seconds

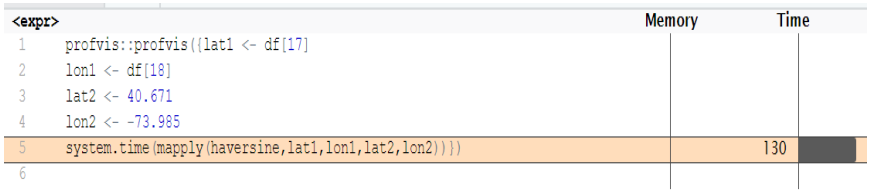
b) Next, replicate the for-loop based approach (the first one) and two different ways to make that version more efficient, in R. Profile these three approaches, and tabulate the results.

Method type	System.time() User system elapsed			Runtime
mapply()	0	0	0	130 ms
for loop	0.02	0.00	0.02	20 ms
Foreach loop	0.02	0.00	0.02	20 ms
apply()	0	0	0	10 ms

Profvis – for loop



Profvis for each loop



profvis – mapply()

Code	File	Memory (MB)		Time (ms)
▶ profvis::profvis		0	0	10

Profvis – apply()

c) Based on the computational efficiency of implementations in Python and R, which one would you prefer? Based on a consideration of implementation (i.e., designing and implementing the code), which approach would you prefer? Taking both of these (run time and coding time), which approach would you prefer? Identify and describe one or two other considerations, in addition to these two, in determining which of the two environments – Python or R – is preferable to you.

After implementing distance calculation on the clinics.xls file, I would prefer Python implementation. I would prefer Python when considering designing and implementation of the code. Python would be the most appropriate language for me in terms of run time and the coding time. I feel much comfortable to write python scripts and it is easier to implement than R code. Python houses rich libraries that are easier to execute and python code is easily readable in comparison to R. Python also provides additions platforms like Biopython, Cython, Anaconda Python. Overall, all these unique properties make python more user friendly. It is also believed that Julia programming language has more pros than python and R.