Baruch “Big Data” Course- Homework Assignment A

Q1- What native features of Python support parallel programming?

Python has a native *Multiprocessing* library that supports parallel programming.

Q2- What native features of R support parallel programming?

The only way to parallelize your entire algorithm or just specific sections that can benefit from parallelism in R is to download a host of R Libraries packages.

Q3- What Python libraries support parallel programming?

Aside from the standard *Multiprocessing* library, a list of libraries (both active and inactive) can be found on the Python Wiki page ( <https://wiki.python.org/moin/ParallelProcessing> ).

Q4- What R libraries (packages) support parallel programming?

The *foreach* library and *doParallel* library support parallel programming in R. There are numerous other libraries that can be found listed in the cran.r-project.org website. Some of the libraries listed are dependent on other libraries. R has a great way of identifying these requirements so that you download all of the libraries needed in order to get your code to recognize the pertinent functions.

Q5(bonus)- Can GPUs be used with Python and R? If yes, then how?

Yes, both Python and R can make use of GPUs. In order to use it in R you will need to download the *gpu*R package. In Python and R, Nvidia’s CUDA GPUs can work together by downloading the right library packages. The appropriate packages can be found on the Python Wiki Page And Cran Project page mentioned before.