

The Power of Efficiency

Camille Heathcock

Efficiency is the principle in programming that all code should be as concise as possible to not only make the code run smoothly, but also to help others to comprehend what it is doing. Efficiency in statistical computing looks like concise, non-repetitive code which uses iteration and modern tools to carry out an analysis, as outlined by the syllabus. It also means not using the same function multiple times. Efficiency allows the programmer to use different inputs to the same code to conduct various analyses. This is enabled by the use of iteration and user-defined functions. Efficiency also helps the programmer and others to be able to read and comprehend code. Efficiency is so important because it creates readable code and supports robust programming. An early ah-ha moment I had related to efficiency is the idea that several arguments can be called in the same function. A great example of this would be from my revised Lab 7, part 2 task 2. When I originally did the Lab, I called the `stopifnot()` function 2 times in a row, but then realized that it would be much more efficient to just separate the 2 arguments within the function with a comma. Another important ah-ha moment I had about efficiency was using the pipe operator. The pipe operator is a great way to create seamless code and minimize interstitial objects having to be created. A good artifact to represent this would also be from Lab 7, under Part one, Task 2. I was able to efficiently filter and select my variables, and then use `ggplot` all within the same pipeline.