Vectorized thinking! Teaching efficient programming in MATLAB

Matin Rahnamay Naeini, Ph.D.
rahnamam@uci.edu
University of California, Irvine

My exposure to computer programming started at high school, where I learned about the basic concepts of programming in Visual Basic. I remember our teacher spent a lot of time clarifying "loops" and "conditional" operators' concepts. Later, as a freshman, I enrolled in a "Basics of programming" course, which again our teacher spent a lot of time to show us how to use these operators in FORTRAN and Pascal. My experience with these programming languages and exposure to that school of thought made these operators an inseparable part of my coding.

Later, I started to use MATLAB for my research, and I had the same school of thought for programming. I used "for loops" for everything and used to control all my variables with as many "if statement" as possible. I was very confident that I learned to code in MATLAB, and I can vividly remember the smile on my adviser's face when I proudly showed him my first code. I was complaining to my friends how slow MATLAB is and I wished I started my research in another programming language. Few month later, I started to dig more into the core idea of MATLAB, I realized that there are other school of thoughts. I got introduced to the concept of vectorization and realized how much life could have been easier if I knew the idea. I realized how fast and clean my codes would by vectorizing my code. However, transition to vectorized thinking (Blitzstein and Hwang 2014) was not very easy in the beginning.

Later that year, I was assisting a faculty for an "Introduction to MATLAB programming" undergraduate level course. I realized that a big portion of the course material revolves around loops and conditional operators, while not much was mentioned about vectorization. Of course, these operators are among the most important concepts in programming and need to be emphasized in any introduction to programming courses. However, vectorization is one of the main strengths of MATLAB, which needs to be emphasized and practiced. In particular, vectorized thinking is an essential skill for any STEM student in the era of big-data and machine-learning to write efficient codes.

References

Blitzstein, Joseph K, and Jessica Hwang. 2014. *Introduction to probability*. Chapman and Hall/CRC.