Adventures in High Performance Computing and R: Going Parallel

Justin Harrington & Matias Salibian-Barrera

Parallel computing refers to the ability to use multiple CPUs concurrently to perform calculations that would otherwise be carried out sequentially in a single CPU. While not all statistical applications can benefit from using parallel computation, those that can are able to scale near-linearly their processing times for increasing numbers of CPUs, all other things remaining constant.

Generally, researchers whose area of interest is not computer science tend to look at parallel computing as something that would require: (a) access to an expensive multiple CPU machine or cluster, and (b) that they completely re-write their code to accommodate this sophisticated architecture. In this talk I will argue that nowadays these concerns should not prevent R users from exploring (and using) parallel computing.

Although it is true that massive computer systems with multiple CPUs are still beyond the means of most individual researchers, technology now exists for establishing virtual "parallel computing" clusters using groups of non-homogeneous computers that sit idle for significant lengths of time (such as those in a teaching lab, for example). This "virtual cluster" can also be constructed using the CPUs inside a multi-CPU machine running a standard OS.

The other important consideration that keeps many researchers away from considering parallel computing is the perceived large overhead cost involved in re-writing their computer code to take advantage of multiple CPUs. Fortunately, this is not necessarily the case, as long as the original R code was written in an efficient way (from a single CPU point of view) because R has several readily available libraries that allow us to use multiple CPUs with minimal changes to a "standard" R program / function.

In this talk I will be discussing how R can take advantage of parallel processing. In particular, I will discuss the libraries `rpvm`, `Rmpi` and `snow` that facilitate migrating "standard" R code to take advantage of multiple CPUs (either in the same motherboard or across the internet). I will also briefly discuss MPI (Message Passing Interface) and PVM (Parallel Virtual Machine), the protocols used to control the architecture.

This talk will be presented from the perspective of a statistician rather than a computer scientist, and the focus will be on helping users get started. I will discuss how to use these libraries and demonstrate their application on one clustering method, Linear Grouping Analysis (Van Aeslt, Wang, Zamar & Zhu (2006)), where this strategy of parallel computing has been successfully applied.