**R or Python: A Programmer’s Response**

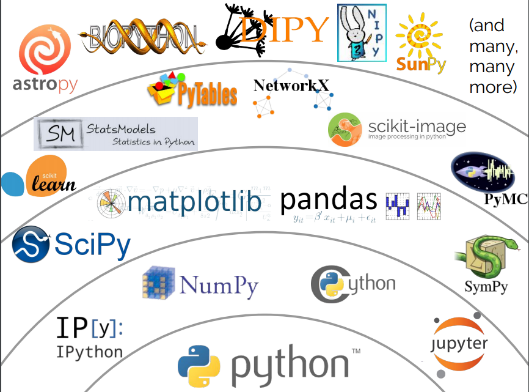
**Abstract/ Instruction:**

We hear lots of requests on porting R to Clinical Computing Platform. Before R really touches the ground, I’d like to share some thoughts on why R is a not a good choice, considering there is a strong alternative, Python.

The requests on R are mostly from statisticians inside clinical development units. Statisticians are not programmers; they might prefer R for some statistical tasks, like sample size estimate and visualization. But for heavy users for the Clinical Computing Platform, aka, clinical/statistical programmers, they need good language(s) to import data, clean data, transform data and do analysis and reporting. In this type of programming, I’d argue Python is far way better than R, from a programmer’s perspective.

First, Python is simply a better language, created by real computer scientists. R was mainly created by statisticians and for statisticians. There are lots of design drawbacks in R core language.

Second, for programming tasks mentioned above, Python offers more consistent and unified packages stack (Numpy, Pandas, Scikit-learn, etc), while in R, the packages are scattered. It’s true that there is an attempt to gather several core R packages to unify the process, but at least at this stage, it’s not satisfactory.



Third, Python is far way better at system integration, with current components residing in Clinical Computing Platform.

Fourth, considering Python’s more elegant syntax and unified ecosystem, I’d sure actually Python is easier to learn for SAS programmers, although it might not be well known in clinical programming world. R, instead, consists of lot of black tricks by statisticians.

Last but not least, Python will make a happier programmer. Python will introduce SAS programmers to a more wider world of machine learning, deep learning which all cool kids are talking about…